REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	[redacted]9	DISSERTATION

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
A COMPERHENSIVE APROACH TO SENSOR MANAGEMENT AND SCHEDULING	
6. AUTHOR(S)	
LT COL MCINTYRE GREGORY A	

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
GEORGE MASON UNIVERSITY	

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
THE DEPARTMENT OF THE AIR FORCE AFIT/CIA, BLDG 125 2950 P STREET WPAFB OH 45433	FY99-25

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Unlimited distribution In Accordance With AFI 35-205/AFIT Sup 1	

13. ABSTRACT *(Maximum 200 words)*

19990120 023

14. SUBJECT TERMS	15. NUMBER OF PAGES
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT

A Comprehensive Approach to Sensor Management and Scheduling

by

Gregory A. McIntyre

A Dissertation
Submitted to the
Graduate Faculty
of
George Mason University
in Partial Fulfillment of
the Requirements for the Degree
of
Doctor of Philosophy
Information Technology

Committee:

Kenneth J. Hintz, Dissertation Director

Andrzej A. Manitius

Stephen G. Nash

Daniel B. Carr

Stephen G. Nash, Associate Dean for Graduate Studies and Research

Lloyd J. Griffiths, Dean, School of Information Technology and Engineering

Date: 09/24/98

Fall 1998
George Mason University
Fairfax, Virginia

A Comprehensive Approach to Sensor Management and Scheduling

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at George Mason University

By

Gregory A. McIntyre

Bachelor of Arts, Washburn University of Topeka

Master of Science, Air Force Institute of Technology

Director: Dr. Kenneth J. Hintz, Associate Professor
Department of Electrical and Computer Engineering

Fall Semester 1998
George Mason University
Fairfax, Virginia

ACKNOWLEDGEMENTS

"Wisdom is realizing that you are going down the wrong path ... again."
Author unknown

First and foremost, I would like to thank my dissertation director and mentor, Dr. Kenneth J. Hintz for his support, advice, never-ending encouragement, and friendship throughout my studies and research. His teaching and research methods were extremely enlightening and proved invaluable in my work. Thanks for your guidance, wisdom, and most importantly the mental health breaks.

I would also like to thank the other members of my dissertation committee, Dr. Andre Manitius, Dr. Stephen Nash, and Dr. Daniel Carr for their valuable advise and suggestions to improve my research.

To my fellow graduate students, I would like to thank you for you help in surviving graduate school and for your friendship. Special thanks go to Shawn Masters, John Bartone, Alaa Sheta, Eddie Mayhew, Nirmal Warke, and Jim Mosora.

Finally, I would like to express my sincere thanks and heartfelt appreciation to my family – especially my parents and my wife. I am forever grateful for your support and tireless

encouragement and your faith in me throughout the years. Tina, we finally did it - now let the good times begin.

Table of Contents

List of Tables

List of Figures

ABSTRACT

A COMPREHENSIVE APROACH TO SENSOR MANAGEMENT AND SCHEDULING

Gregory A. McIntyre, Ph.D.

George Mason University, August 1998

Dissertation Director: Dr. Kenneth J. Hintz

Heterogeneous multisensor systems have been widely used in a variety of military and civilian applications. While the majority of research in multisensor systems is dedicated to military applications, other applications include robot navigation, autonomous vehicles and paramilitary operations. In general, single sensor systems only provide partial information on the state of the environment while multisensor systems rely on data fusion techniques to combine related data from multiple similar and/or dissimilar sensors. The goal of a multisensor system is to provide a synergistic effect that enhances the quality and availability of information about the state of the world over that which would be acquired solely from one sensor.

Sensor management can be described as a system or process that provides automatic or semi-automatic control of a suite of sensors or measurement devices. Previous approaches to sensor management all appear to suffer from the mixing of sensor physical requirements with information needs. The result has been *ad hoc* point solutions that treat the problem as a single

optimization task with a performance measure as a weighted sum of diverse, noncommensurate measures. This dissertation presents a new mathematical representation of the multisensor system to capture the sensor management process. Based on this representation, an original hierarchical sensor management model is developed that partitions the system into its constituent processes. These include the sensors themselves, the targets, the Fusion Space, and the Information Space. The Information Space is further partitioned into the Mission Manager, the Information Instantiator, and the Sensor Scheduler.

Additionally, this dissertation describes a new approach which uses partially ordered sets to construct a goal-lattice that converts qualitative mission goals to quantitative values for different sensor actions. This approach superimposes value apportionment on the lattice in order to provide a mathematically quantitative and traceable measure of importance (weights) that a sensor manager can use to optimize trade-offs among competing management functions to meet the mission goals. Another advantage is that these weights can vary as a function of time or phase of a mission thus providing a mathematically based methodology to modify the preferences in real-time based on changes in information produced by data fusion, a human operator, or both.

Chapter 1

Introduction

1.1 Motivation and Problem Definition

Heterogeneous, multisensor systems (referred to hereafter simply as multisensor systems) have been widely used in a variety of military and civilian applications. While the majority of research in multisensor systems is dedicated to military applications, other applications include robot navigation [1], [2], [3], autonomous vehicles [4], [5], [6], and paramilitary operations (*e.g.* drug interdiction [7]). In general, single sensor systems only provide partial information on the state of the environment while multisensor systems rely on data fusion techniques to combine related data from multiple similar and/or dissimilar sensors. The goal a multisensor system is to provide a synergistic effect that enhances the quality and availability of information about the state of the world over that which would be acquired solely from one sensor.

Until recently, sensors were fewer in number and less capable than they are today. An operator could easily decide which sensor to use, when to use it, point and control it, and even how to interpret the data. Even the environment in which these systems were used was simpler with fewer and less diverse threats. However, the performance characteristics of modern sensor systems have improved dramatically resulting in more able and diverse systems [8]. These improved performance characteristics include [9]:

– All weather

– Jam resistant

– Large search areas

– Emission control

– Improved accuracy

– Aperture agility

These technological advances and the use of multisensor systems have also led to a tremendous increase in the amount of data requiring processing. The number, types, and agility of sensors along with the increased quality and timeliness of data have far outstripped the ability of a human to control them. With all of the different types of sensor and noncommensurate data, it is often difficult to compare how much information can be gained through a given sensor scheduling scheme. This has resulted in a need for an automated sensor management system that optimally schedules the selection and use of individual sensors from among the several available in the system.

Sensor management can be described as a system or process that provides automatic or semi-automatic control of a suite of sensors or measurement devices in a dynamic, uncertain environment. In general, it is the sensor manager that must determine [10]:

– Which service?

– What sensor?

– Where to aim?

– When to start?

while monitoring sensor performance. At its simplest level, a sensor management system is a control process that must deal with [11]

- Insufficient sensor resources
- Highly dynamic environment
- Varied sensor capabilities
- Varied sensor performances
- Randomly occurring sensor failures and
- Enemy interference and spoofing

Thus, a sensor manager is expected to [8]:

- Reduce the operator workload by automating sensor allocation
- Prioritize measurement requests to meet both integrated flight management and weapons control requirements
- Aid data fusion by coordinating information requests with sensor observations
- Support sensor reconfiguration and degradation due to partial or total loss of a sensor

However, sensor management is only one part of the overall process. One paradigm used to explain the use of sensors and sensor management is shown in Figure 1-1. The key component of this paradigm is information -- specifically how to optimally obtain information about the state of the environment through the application of sensors. Van Creveld [12] states that "The history of command in war consists essentially of an endless quest for certainty about the state and intentions of enemy forces ...". It is in the data fusion portion of the command and control system that sensor measurements of the environment are processed in order to reduce the

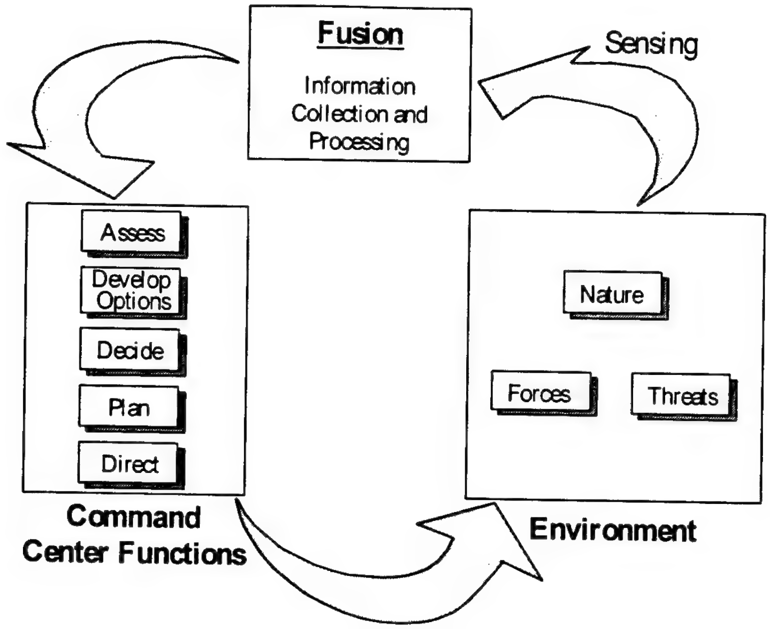


Figure 1-1: Command and Control Paradigm

commanders' uncertainty about the environment. More specifically, "...data fusion is a process dealing with the association, correlation, and combination of data and information from multiple sensors and sources to achieve refined position and identity estimation and complete timely assessments of situations and threat, and their significance [13]."

A conceptual depiction of the overall process flow is shown in Figure 1-2. The environment is comprised of a set of targets and their states. These target states can be divided into two subsets - those targets that have not been detected and those that have been detected and are, or will soon be, in track. Those targets that are in track can be further subdivided into two subsets - targets that have been identified and those that have not been identified. Sensors under the control of the sensor manager make measurements of the physical phenomenon exhibited by the targets and combine these into observations. These observations are then processed in order to

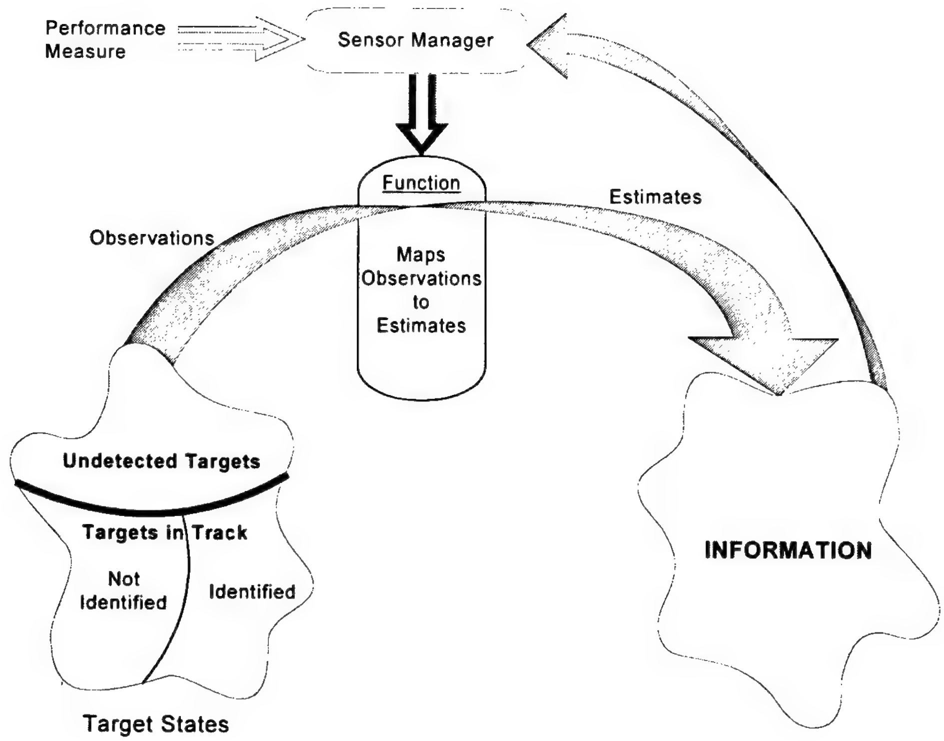


Figure 1-2: Process Overview

provide target state estimates. These estimates are then combined with other sensory data and external inputs and inferences to obtain information. The sensor manager then uses this information along with some, not necessarily time-invariant, performance measure to control the next measurements made by the sensors. What these two paradigms show is that sensor management is a control process and data fusion is an estimation process. It also highlights the fact that both processes are interrelated.

1.2 Sensor Management Applications

The impetus for this research is based on what is called the "in harm's way" mission of a surveillance aircraft. The aircraft is capable of carrying several different types of sensors (*e.g.* infrared, radar imaging systems, and electronic surveillance measures). The aircraft is sent out

on a surveillance mission with or without any *a priori* information about the target environment that it is to operate in. The information sought here is state information about any potential threats (targets). The goal of the mission is to detect, track, and identify as many targets as possible. Several other well defined military applications are presented by Musick and Malhotra [9] and Malhotra [14]. Essentially, the Sensor Manager's task is to provide the most effective transfer of information from the real world to our internal mathematical model of the world. That is, subject to operational constraints, it is desired to minimize the mean-squared error between the actual and estimated target state (both kinematic and nonkinematic) through the allocation of sensing resources.

While most of the research in sensor management has been directed towards tactical military applications, sensor management is not limited to this application. Two other examples include the search and rescue of individuals in hazardous situations and the management of several low earth orbit satellites to maintain space object ephemeris. There also appears to be applications of this approach to data mining in large databases. The search and rescue example is part of NASA's effort to develop and apply aerospace technologies capable of locating aircraft, ships, spacecraft, or individuals in potential or actual distress and then provide immediate aid to extract victims to safety. While this NASA effort spans a wide range of disciplines, sensor management can also be applied to remote sensing. Specifically, sensor management is required to manage the wide variety of sensor (foliage penetrating synthetic aperture radar, laser systems and multi- and hyper-spectral optical scanners) to detect and identify small targets and optimize tactics for search using remotely sensed data. The satellite example is part of an ongoing project for space object surveillance involving approximately 30 low earth orbit satellites with severely

constrained viewing geometry. There are currently about 8000 objects (satellites, space junk, *etc.*) in orbit with a projected 15,000 objects shortly after the turn of the century. The constraints placed on the sensor aboard each satellite include:

– Extremely limited field of view due to the requirement to image the object against deep space

– Can't image against bright background (*e.g.* sun, moon, earth)

– Can't imaging in the Earth's shadow

– Track time duration is approximately 25 seconds (varies with target)

Additionally, the viewing opportunities also vary in quality and are dependent on

– Viewing angle (better of larger angle)

– Distance from sensor to target (quality decrease with increase in distance)

– Sun angle and object reflectivity (effects object brightness)

– Target and satellite movement during view

1.3 Major Contributions

A variety of partial (open-loop) sensor management approaches have been proposed (and will be reviewed in Chapter 2), all of which appear to suffer from the mixing of sensor physical requirements with information needs. This commingling of inappropriate, noncommensurate measures leads to *ad hoc* methods of sensor management and no comprehensive framework in which to develop the separable components of a complete system. The *ad hoc* nature of these solutions, essentially "point solutions," does not allow for direct comparison, evaluation, or evolutionary improvement.

The research presented in this dissertation proposes a new approach to sensor management based on information theoretic measures and lattices of partially ordered sets (POSET) along with a new, comprehensive hierarchical sensor management model. Expected information gain, as measured by the expected change in entropy, has been shown to be a valid approach to sensor management for determining the trade-offs between search, track, and identify. While using this measure of information gained is a necessary condition, it is not a sufficient condition for complete sensor management. That is, if one uses only information gained as a means to perform sensor management trade-offs, it does not take into account the multiplicity of competing mission goals.

The approach developed and presented in this research to overcome this limitation is the use of inclusion relationships among goals and partially ordered sets of these goals. This facilitates the construction of a hierarchy of goals and a mathematical means to weight the multiple, competing goals. The result is, regardless of the type of scenario – military or civilian, a method that results in a new, quantitative, and traceable measure of importance that a sensor manager can use to perform and optimize trade-off among search, track, and identify information needs.

This hierarchical, reductionist sensor system introduced here maintains its own representation of the world or environment at different levels of abstraction in different levels of the hierarchy. The highest level in the hierarchy incorporates mission requirements and human inputs to determine the values or relative preferences among search, track, and identification as quantified by a weighted, non-stationary, lattice of goals. The next level contains the function of information management in the form of an information-to-observation mapper referred to as an

Information Instantiator (II) which converts an information need into an observation function (described in greater detail in Section 2.4). The actual allocation of this observation function to a specific sensor or set of sensors which make the measurements is optimized in the next level by a separate sensor scheduler. The optimization criteria of the sensor scheduler are based on sensor related concerns as well as priorities assigned to observation tasks by the Mission Manager (MM) and passed to it by the Information Instantiator.

A fundamental task of the information space, at least the sensor manager portion of it, is to provide the most effective transfer of information from the world to our internal mathematical model of the world subject to operational constraints and in consonance with a time-varying set of ordered goals. One component of that process is the conversion of information needs required to search for new targets, maintain targets in track, and identify targets in track. A second component is the manner in which the mission manager dispatches requests to the Information Instantiator. The details of the quantification of goals and the determination of the relative value of search, track, and ID is covered in Section 2.5.

Since the terminology is not generally agreed upon, for the purposes of this dissertation, the following definitions will be used and are consistent with the definitions in the Oxford English Dictionary [15]. *Information* is a change in uncertainty about something. An *observation* is the estimation of a property of a target through the mathematical combination of one or more measurements, possibly combined with other data. A *measurement* is the fundamental acquisition of data about a target through the use of some physical property of the target (*e.g.*, reflected energy) or environmental property caused to change by the target (*e.g.*, wake).

The information space can be thought of as operating as the integration of two different concepts. The first is *a touch it once* approach. *Touch it once* implies that when an event occurs, the mission manager decides what to do, dispatches the task, remembers that it has done so by putting the task on a queue, and waits for another event (which may be an internal need as well as an externally driven event). The second concept is that of a discrete event simulation (DES) in which a queue of events to be performed at a later time is maintained. The events referred to here are the tasks which have been passed to the II but are also put on MM's queue indicating measurements scheduled to be executed in the future. Entries in this queue contain such data as which contact number, what kind of information is needed, when is the information needed, how much information is needed, and why this event was scheduled.

Going back to the "in harm's way" example, assume that the mission manager has no *a priori* information about its target world. Of the three choices, neither tracking nor ID is appropriate, so search is the only alternative. The search task is dispatched to the sensor manager which searches utilizing all the sensors until a target is detected. This detection generates an event with which the MM must now deal. Since it is the result of a search, the MM can now choose to either identify or track based on the relative value of the two options as determined by the non-stationary goal lattice. The Mission Manager then responds to the event by dispatching to the II portion of the sensor manager a request for ID or track data from the target. This request includes the goal-derived value of the observation from the lattice as well as the required accuracy and temporal constraints and places it in its own queue of dispatched requests. This is essentially a request for information without indicating how it is to be satisfied. The MM is no longer concerned with the queued task unless or until the II and/or the Sensor

Scheduler reply to the request with either the result of the observation or an *unable-to-observe* acknowledgment. It is important to note that an *unable-to-observe* response may be caused by too restrictive a requirement on the information, *i.e.*, a request for the information too soon or at too low of a priority to preempt other executing or scheduled sensor manager tasks.

1.4 Dissertation Overview

Chapter 2 provides a comprehensive description and literature review of the state-of-the-art sensor management including a discussion of a newly proposed, mathematically rigorous sensor management model of a multisensor system. Also discussed is the formulation of a new and original comprehensive model which is used to develop an information theoretic approach to sensor management combined with the use of partially ordered sets to compute weights or values of different sensor functions in order to facilitate the trade-offs between them.

The goal of the research presented in this dissertation is to apply measures of information to managing multiple sensors in order to obtain a near optimal real-time sensor utilization within the constraints of the mission requirements and sensor limitations. Chapter 3 briefly reviews the use of information theory within the context of sensor management including a description of the measures that will be used for calculating the expected information gain for search, track, and identification. It further discusses why information is chosen as the measure to maximize.

Kinematic state estimation is an important aspect used within a sensor management system. As such, a review of maneuvering target tracking is presented in Chapter 4 along with a comparison of several Kalman filter models used in the target kinematic state estimators.

The simulation of the proposed sensor management model is presented in Chapter 5 and the results are presented in Chapter 6. Finally, Chapter 7 provides a summary of the work presented here and concludes by discussing the strengths and limitations of this proposed sensor management model along with possible future research.

Chapter 2

Sensor Management and Sensor Scheduling

2.1 The Sensor Management Role

As discussed in Chapter 1, technological advances and the use of multisensor systems have led to a tremendous increase in the amount of data being processed that has far outstripped the ability of a human to control it. The data provided by different sensors is of different units, dimensions, and types (detections, position, or target class or subclass). With all of the different types of sensors and this noncommensurate data, it is often difficult to compare how much information can be gained through a given sensor scheduling scheme. This has resulted in the need for automated sensor management systems that optimally schedules sensor measurements.

Often the terms sensor management and sensor scheduling are used interchangeably but they are not the same. Sensor management can be defined as "...the process which seeks to manage or coordinate the use of sensing resources in a manner that improves the process of data fusion and ultimately that of perception, synergistically [16]." This reduces to an almost trivial situation if there is only one sensor or no contention for sensor resources. Sensor scheduling refers to the actual allocation of measurement tasks to specific sensors. Figure 2-1 depicts the evolution in sensor management research and development (R&D) as characterized by Denton, *et al.*[10] As sensor systems and the associated computers and signal processing techniques

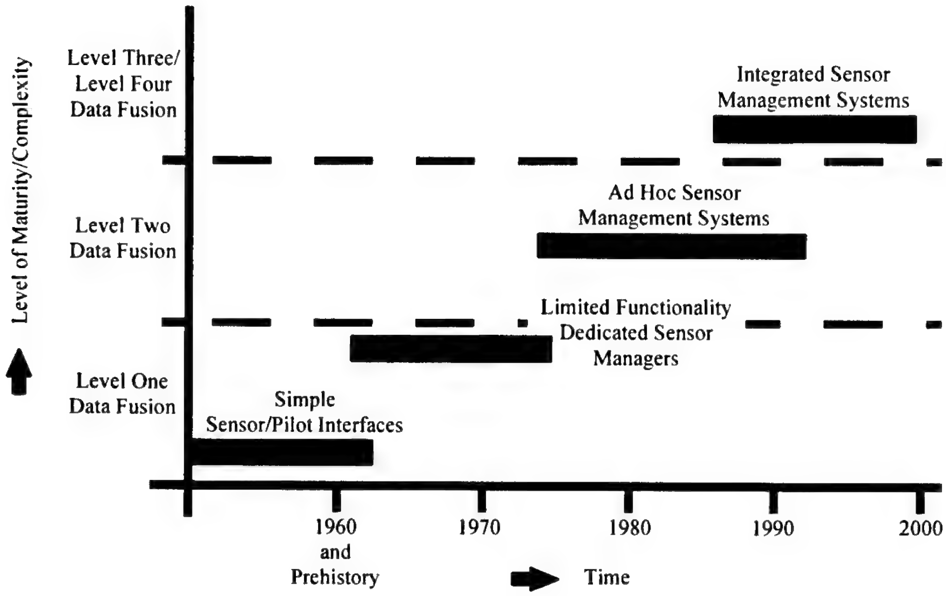


Figure 2-1: R&D in Sensor Management Systems [10]

improved, the levels of data processing also evolved. Sensors of the 1960's era were simple enough that the pilot performed both the sensor management and data fusion functions himself. Present-day sensors are more agile and more numerous resulting in an increase in the amount of data being processed. "As data quantities increase and control choices multiply, workload increases exponentially and eventually even the most able pilots begin to miss important opportunities or fail to recognize critical situations [9]." This has resulted in the need for integrated automatic or semi-automatic sensor management systems.

Popoli [17] describes sensor management as a feedback control system. The system attempts to obtain the most information from the available sensors by continually monitoring the sensors' performance. This is done by comparing performance relative to a specified criterion (see Rothman and Bier [18] for a comprehensive list of performance measures). This generates feedback control to the sensors.

Sensor management is important in terms of the benefits it provides over non-coordinated sensor operation. By automating the process, it reduces the operator workload. The operator defines the sensor tasking criteria instead of controlling multiple sensors individually by specifying each operation to be performed by each sensor. In an automated, semi-autonomous sensor management system, the operator concentrates on the overall objective while the system works on the details of the sensor operations. This allows for multisensor fusion by taking advantage of the strengths of each sensor. Additionally, the feedback within the sensor management system allows for faster adaptation to the changing environment. Thus the sensor management system effectively uses the limited resources available [17].

The representation of the sensor management function and its relationship to data fusion developed and used in this research is shown in Figure 2-2. Sensors are tasked to make measurements of the environment. These measurements are then processed to obtain observations and then combined to obtain information. This information is used by the Mission Manager (along with internally or externally generated performance measurements) to generate information requests to be processed by the Sensor Manager. The Sensor Manager is partitioned into two orthogonal functions, one concerned with the information to observation request mapping (Information Instantiator) and the other concerned with mapping these observation requests to sensors measurement requests or tasks (Sensor Scheduler). It is important to distinguish between the functions performed by the Information Instantiator and the Sensor Scheduler.

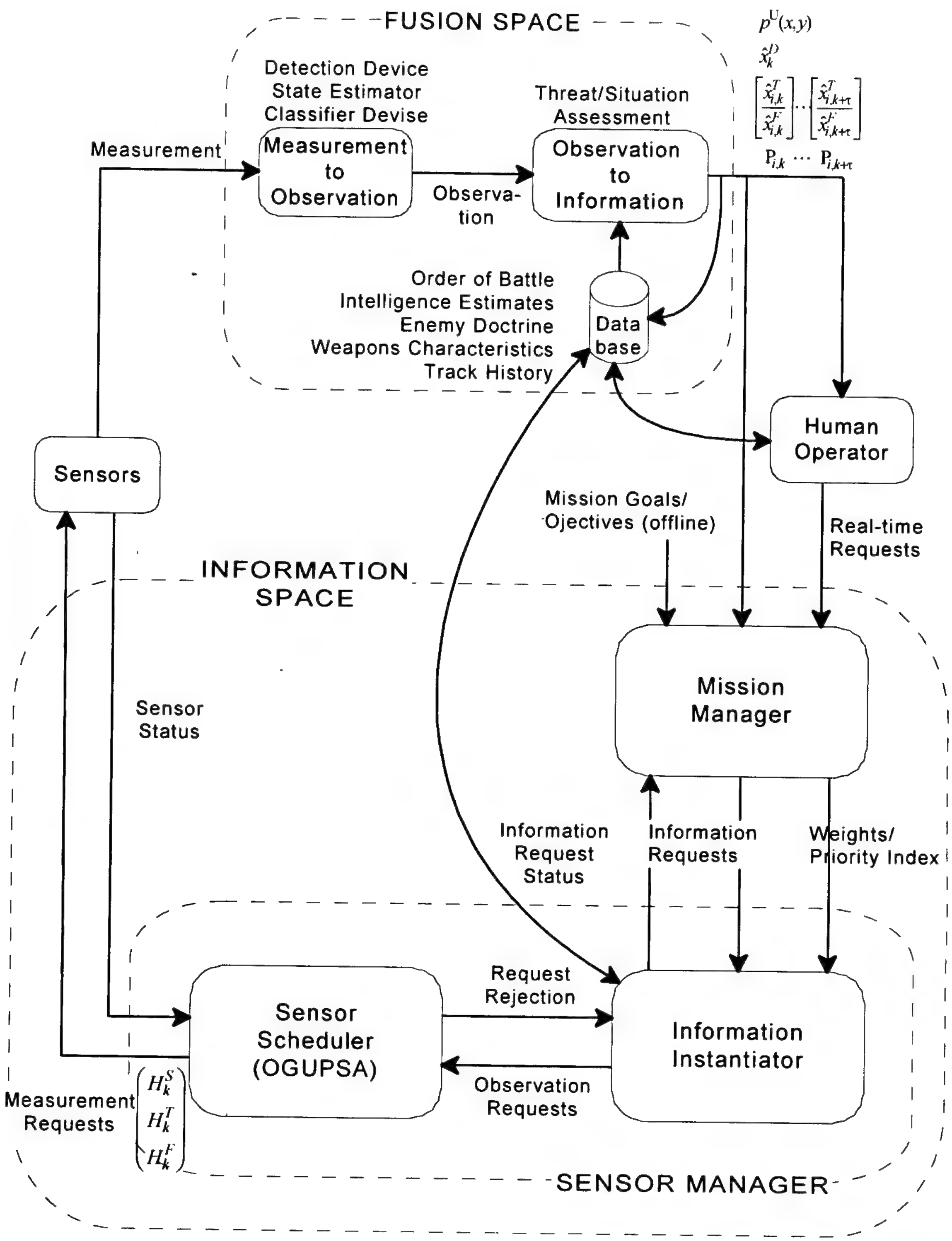


Figure 2-2: A Sensor Management System Showing the Relationship Between Data Fusion and Sensor Management

While the Mission Manager is concerned with metasensor issues such as:

–How accurately to measure?

–Which service to perform (*e.g.*, search, track, fire control, *etc.*)?

–From what physical location of the environment to obtain a measurement?

–When is the earliest usable time to begin the measurement?

–What is the latest usable completion time for the measurement?

the problem for the Information Instantiator is to determine how to maximize the effectiveness of individual sensors or a collection of sensors while simultaneously optimizing such conflicting objectives or goals as

–Detection

–Tracking

–Identification/Classification

–Emission control (EMCON)

In contrast, sensor scheduling deals with intrasensor which include:

–Which sensor or combination of sensors can best perform the measurements required of a observation task?

–How do sensor interact (*e.g.* radar interfering with ECM)?

–Which sensor mode?

–What scan volumes, beam scheduling and/or dwell-time?

In order to determine how to accomplish a list of tasks based on sensor availability and capabilities, Zhang and Hintz [19] developed an on-line, dynamic, preemptive sensor scheduling

algorithm called the On-line, Greedy, Urgency-driven Pre-emptive Scheduling Algorithm (OGUPSA). McIntyre and Hintz enhanced [20] the OGUPSA algorithm and demonstrated its use in a sensor management simulation [21].

2.2 Requirements, Functions, Principles, and Problems

The goal of sensor management is to perform the right task at the right time on the right object based on external performance measures or criteria. This is a complex task considering that the sensors must work within a highly dynamic, nonstationary environment and with finite sensor capabilities and availabilities. It is important to note that the sensor manager is trying to optimize the utilization of a finite set of sensors with a finite computational capability in this dynamic, non-stationary environment to maximize the flow of information about the environment so that a mission (goal) can be successfully completed (achieved). As a result, the sensor manager must [8]

– Permit maximum flexibility for sensor resource allocation

– Maintain mission effectiveness in a degrading hardware environment

– Possess maximum self monitoring capability

– Exhibit minimum response time while servicing many near-simultaneous requests

while the primary functions of sensor management are:

– How accurately to measure?

– Which service to perform (*e.g.*, search, track, fire control, *etc.*)?

– From what physical location of the environment to obtain a measurement?

– When is the earliest usable time to begin the measurement?

– What is the latest usable completion time for the measurement?

The general principles involved in a sensor management system to effectively accomplish the above functions include [9]:

–Plan to use all sensors (offensive & defensive)

–Value long-term goals of survival and success, not just accuracy and identity

–Dynamic environment dictates adaptive methods

–Choose a modeling technique that is mathematically sound, comprehensive, and tractable

–Account for dissimilarities in sensor ability

–Eliminate redundant sensor allocations and take advantage of sensor synergies

–Provide for emission controls (passive and low probability of intercept modes)

–Achieve iteration rates in planning that keep pace with all environment changes

–Shed load gracefully when sensor burden hits limits

–Consider adaptive-length planning horizons.

The problems that must be dealt with by a sensor management systems include [11]:

–Insufficient sensor resources

–Highly dynamic environment

–Varied sensor capabilities

–Varied sensor performances

–Randomly occurring sensor failures and

–Enemy interference and spoofing.

2.3 Sensor Management Techniques

A variety of techniques have been proposed or applied to the area of sensor management. Buede and Waltz [22] discuss several issues that have been proposed. They include heuristic or

rule based systems with greedy search algorithms; optimization techniques that include decision theory or utility theory, linear programming, and fuzzy set theory, and team theory. Musick and Malhotra[9] review recent applications which include artificial neural networks, decision theoretics, information theory, and mathematical programming techniques including Linear, Nonlinear, and Dynamic Programming. Several other authors [17], [23], [24], [25], [26], [27], [28] address the use of Knowledge-based systems or expert systems.

One of the first articles to apply optimization techniques to sensor management is by Nash [29] in which he uses linear programming to determine sensor-to-target assignment for targets being tracked. Nash uses the trace of the Kalman filter error covariance matrices as the costs coefficients in the objective functions. Also, he uses the concept of pseudo-sensors to handle slack sensor assignments for the case when there are fewer targets then sensor tracking capability. Fung, Horvitz, and Rothman [30] develop a decision theoretic sensor management architecture based on Bayesian probability theory and influence diagrams. Manyika and Durrant-Whyte [31] use a decision theoretic approach to sensor management in decentralized data fusion while Gaskell and Probert [32] develop a sensor management framework for mobile robots also based on a decision theoretic approach. Malhotra [14] discusses the temporal nature of sensor management and describes the sequential decision process as a general Markov decision process. Dynamic Programming is a method for solving a Markov process except that it is a recursive algorithm that determines minimum costs based on the final state and works backwards. Due to this requirement to know, *a priori,* the optimal cost at each stage and the possible combinatorial explosion in enumerating each possible actions in a Dynamic Program, Malhotra proposes using Reinforcement Learning as an approximate approach to Dynamic

Programming while Washburn, *et al.* [33] present a sensor management approach based on Dynamic Programming to predict the effects of future sensor management decisions.

Two optimization approximation approaches applied to sensor management in the literature include the use of fuzzy reasoning and artificial neural networks. Molina López, *et al* [34] present a sensor management scheme that accomplishes sensor tasking using knowledge-based reasoning and fuzzy decision theory. Zhongliang, Hong, and Xueqin [35] use a back propagation neural network to track maneuvering targets over a wide range of conditions. Their target tracking scheme utilizes parallel Kalman filters and uses the neural network to improve position, velocity and acceleration tracking precision. Brownell [36] applies neural networks for sensor management and diagnostics in a production plant to increase energy efficiency while reducing waste and pollution.

Several recent papers have been investigating the application of Information Theory in order to develop a metric that a sensor management system can use to perform sensor-to-task trade-offs. Information Theory, in the form of changes in entropy, has been used in a variety of applications. The most widely used measure of uncertainty is entropy but others include maximum entropy probability estimation, discrimination information functions, and mutual information functions. Hintz and McVey [37] first proposed the use of an information theoretic measures in scheduling a single sensor to track multiple targets. They describe situations where there is either insufficient computation power to utilize all of the available data or where there are fewer sensors than processes to measure. Their approach is to treat the sensors as constrained communications channels and compare them to Shannon's [38] measure of

information capacity in a bandlimited channel. The basis of their approach assumes that the channel is already being used to its maximum capacity in a coding sense, and that more information about the states of multiple processes can still be obtained by choosing that process to measure which will yield the greatest decrease in its uncertainty. Using this analogy, they use the expected change in entropy (as measured by a norm of the error covariance matrix) as a measure of expected information gained for determining which target state estimates to update. This measure is used to maximize the amount of information at each sample interval. Hintz [39] then expands the use of this measure to the cueing of automatic target recognition systems. The result of these two papers is that they place search, track, and identification measure of information into a commensurate space. McIntyre and Hintz [40] use this entropy based information theoretic metric to perform search versus track trade-offs in a simulation program.

Another Information Theoretic approach presented in the literature uses discrimination gain which is based on the Kullback-Leibler discrimination information function [41]. Schmaedeke [42] uses discrimination gain as the cost of sensor allocation in Nash's Linear Program objective function to determine the sensor-to-target tasking. While he shows how this optimally schedules sensors at each time increment, the Linear Program does not run fast enough for real-time applications. Kastella [43], [44] and Schmaedeke and Kastella [45] apply discrimination gain to determine the resolution level of a sensor for measurement to track association. Lastly, discrimination gain is used by Kastella [46], [47] and Kastella and Musick [48] to determine where to search for, and then track, targets based on discrete detection cells representing the probability of a target being present in a cell first for static targets and later for moving targets.

Table 2-1: General Sensor Management References

Focus of Article	Author
Discusses performance criteria	Rothman and Bier [18]
Defines sensor manager requirements and functions	Denton, *et al.* [10]
Defines sensor management, its need, how to accomplish it, and benefits from its use	Popoli [17]
Describes sensor management role in sensor fusion	Waltz and Llinas [49]
Discusses sensor management issues	McBryan, *et al.* [22]
Research at British Aerospace	Upton and Wallace [50]
JDL fusion model including sensor management as Level IV	White, *et al.* [51]
Drug interdiction	Chong and Liggins [7]
Compares several management techniques to detect and classify targets	Kastella and Musick [48]
General discussion of sensor management	Musick and Malhotra [9]
Drug Interdiction/Theater Surveillance	Liggins and Bramson [52]
Tactical Aircraft	Marsh, *et al.* [53], [54]
Manufacturing Robot	Lynch and De Paso [55]

A good summary of data fusion (1) and sensor management (2) and the fundamental issues that they must address is provided by Manyika and Durrant-Whyte [16]. The authors state what the issues are and I quote

1. How can the diverse and sometimes conflicting information provided by sensors in a multi-sensor system, be combined in a consistent and coherent manner and the requisite states or perceptual information inferred?
2. How can such systems be optimally configured, utilised and coordinated in order to provide, in the best possible manner, the required information in often dynamic environments?

The techniques used in sensor management along with their applications are categorized and presented in two tables. Table 2-1 lists several general discussion references with a description of the main focus of the article while Table 2-2 presents a list of techniques and applications

Table 2-2: Summary of Sensor Management Techniques and Applications

TECHNIQUE	APPLICATION
Heuristic	Tactical aircraft [56], [57] Tactical aircraft [58] Rohde and Jamerson [59]
Expert System Multiple Experts Architecture	Tactical aircraft [17], [23], [24] Surveillance networks [25] Tactical navigation [26] ESA Radar control [27] Air defense [28]
Utility Theory	Tactical aircraft [8] ESA Radar scheduling [60]
Automatic Control Theory	Tactical aircraft [61]
Fuzzy Logic/Theory Fuzzy Decision Trees Fuzzy Reasoning	 Tactical aircraft [17] Tactical aircraft [34]
Cognition	Command, Control, Communications [62]
Decision Theoretic Bayesian Belief Networks	Tactical aircraft [30] Mobile robot [31] Mobile robot [32]
Probability Theory Bayesian Approximation Dempster-Shafer Evidence Theory	 Robotic sensor estimation [63] Mobile robot [64]
Stochastic Dynamic Programming Reinforcement Learning	Tactical aircraft [33] Tactical aircraft [14], [65]
Linear Programming	Sensor to target assignment optimization [29], [42]
Neural Networks	Production plant control [36] Tracking maneuvering targets [35]
Genetic Algorithms	Scheduling for sensor management [66]
Information Theoretic Shannon entropy Shannon entropy Shannon entropy Shannon entropy Shannon entropy Shannon entropy Kullback-Leibler/Discrimination Gain Kullback-Leibler/Discrimination Gain Kullback-Leibler/Discrimination Gain Kullback-Leibler/Discrimination Gain	 Military communications, Multiprocess Control, Human supervisory control [37] Sensor cueing [39] Drug interdiction/Theater surveillance [67] Mobile robot [16], [68] Search versus Track trade-offs [21], [40] Sensor management in a decentralized sensing network [69], [70] Sensor to target assignment optimization[42] Tactical aircraft [71] Target detection and classification [46], [47], [48] Multitarget tracking [45]

presented in the literature . The major drawback to most of the references in this survey is that they tend to be point solutions with no mathematical framework for describing, predicting, and comparing performance among various alternatives. This leads to the proposal of a new mathematical model in the next section.

2.4 Mathematical Model

Given any multisensor system, sensors make measurements of the environment. These measurements are combined into observations, and possibly combined with other data to form estimates. These estimates are then combined to produce information. It is this information that is used along with performance measures to control sensor tasking. While this description captures the control and estimation process and provides a satisfactory explanation of the interaction between sensor management and data fusion, there are other issues and components in this process that must be considered. The overriding issue is the consideration of the temporal relationships involved in the process. The other components include search, track, and identification techniques. This leads to the need for a mathematically well-formed, computationally efficient, and near-optimal comprehensive sensor management system.

The formulation of the comprehensive sensor system model presented here is inspired by Malhotra's general analytical model [14] but is mathematical representation that is directly applicable to sensor management. The model is shown in Figure 2-3 complete with all of the processes - search, track, identification, the fusion space, and the information space (which contains sensor management). The term "space" used in fusion and information space is based on the definition of a space as defined by James and James [72] as "Any set or accumulation of

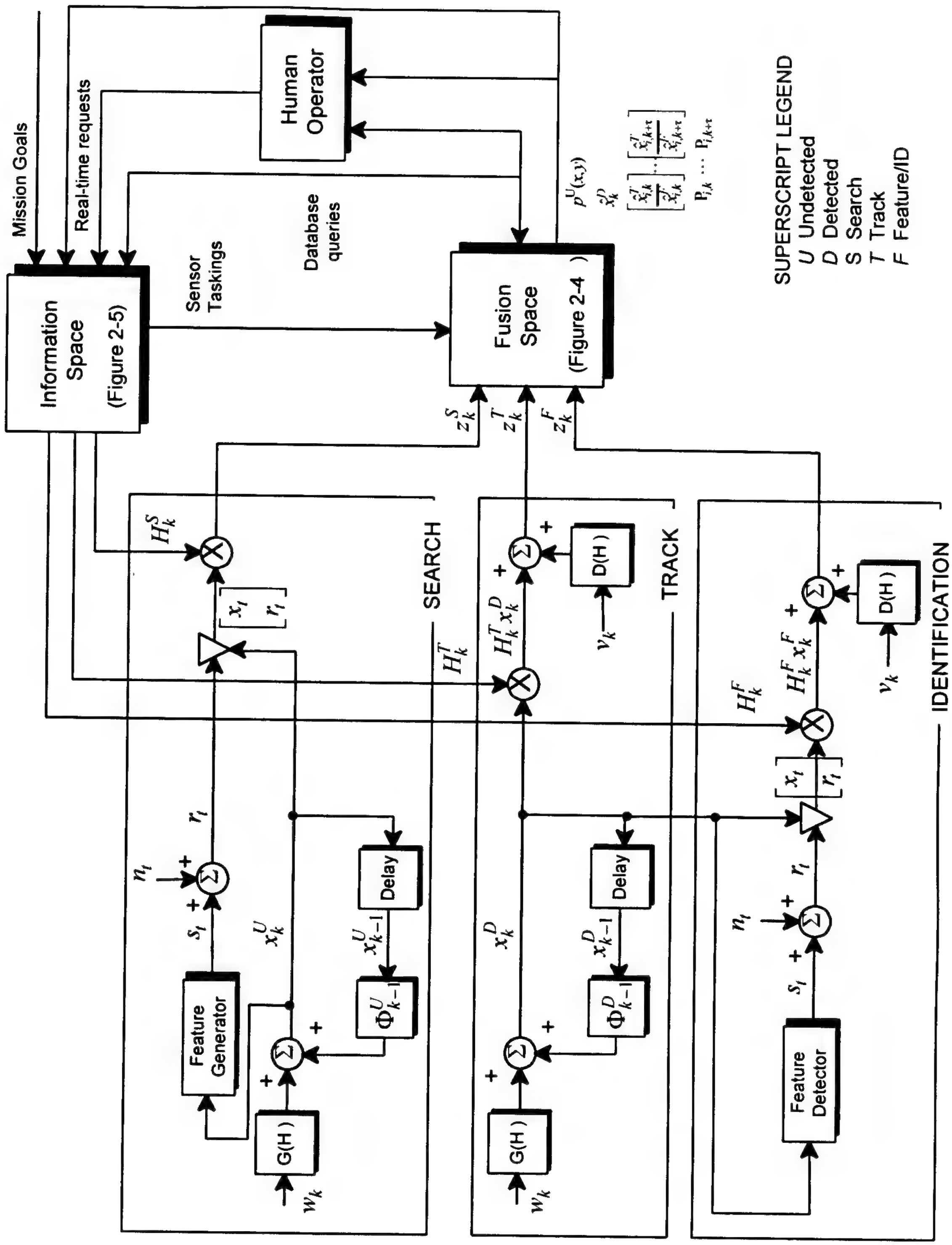


Figure 2-3: Mathematical Model of Target World and Sensor Manager

things, the members being called elements or points and usually assumed to satisfy a set of postulates of some kind." More specifically, they are metric spaces which is defined by James and James as "A set T such that to each pair x,y of its points there is associated a nonnegative real number called their *distance*, which satisfies the conditions:

1) $\rho(x,y) = 0$, iff $x = y$
2) $\rho(x,y) = \rho(y,x)$
3) $\rho(x,y) + \rho(y,z) = \rho(x,z)$

   The function $\rho(x,y)$ is said to be a **metric** of T."

The postulates of the detection space are not covered here but the postulates of the information space include

1) Entropy being a measure of uncertainty
2) Change in entropy is equivalent to change in information
3) Total information available at a given time is measurable
4) Total information available if all processes were to be observed at a given time is measurable and provides an upper bound, and
5) Expected information gain for a given scheduled sensor task is measurable.

These postulates will be discussed in further detail in the subsequent chapters. The fusion space, which contains the data fusion process, is shown in more detail in Figure 2-4. An expanded description of the information space comprised of the mission manager and sensor manager is shown in Figure 2-5.

The target models shown in Figure 2-3, both detected and undetected, are represented with the identical discrete event model. For the undetected target case, the true model is not known to

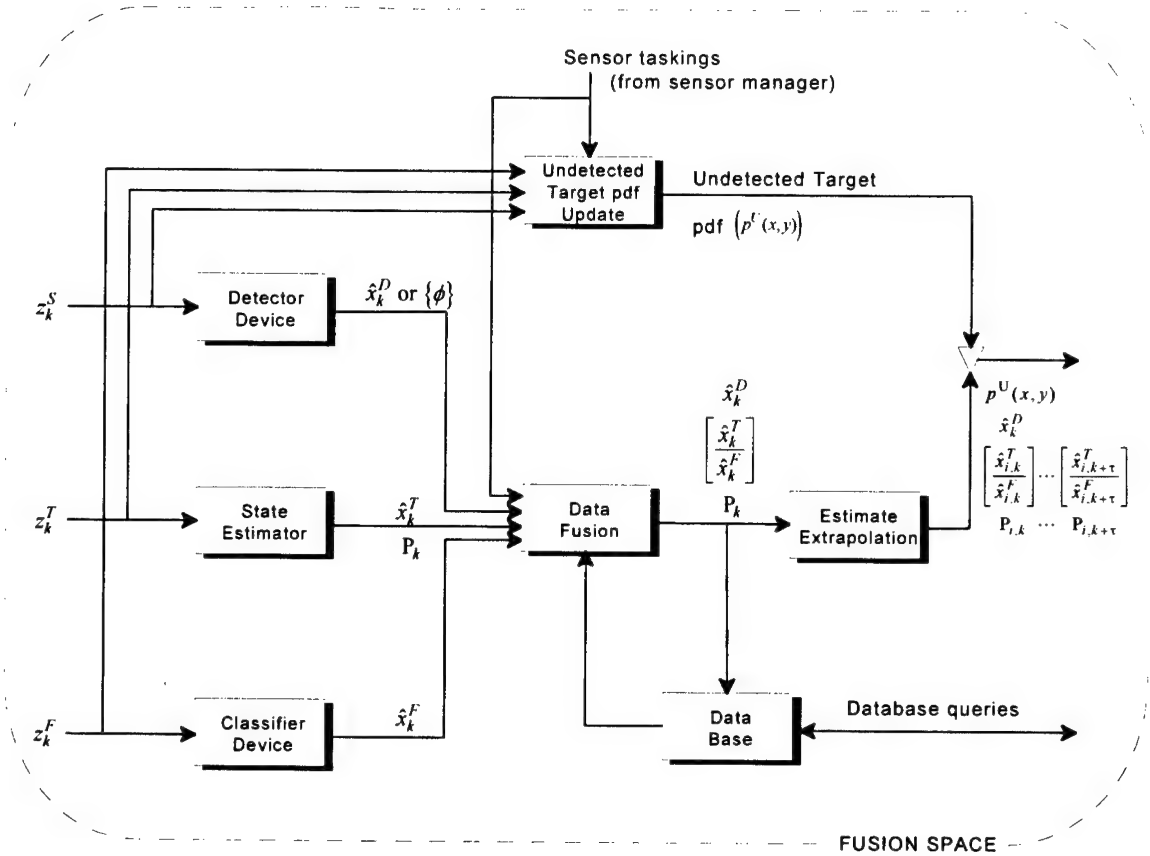


Figure 2-4: Fusion Space Block Diagram

the sensor manager but the estimator target model does influence target detection during the search process. Once a target has been detected (but not yet identified) and is in track the same model which has been used in the search process is used in the tracking portion of Figure 2-3. In the case of target tracking, one can decide between decentralized or centralized estimation and fusion. Also a variety of tracking methods such as innovations-based adaptive filtering, multiple model approach, and image-based direct maneuver estimation can be used [73].

As can be seen in Figure 2-5, the Mission Manager and the Sensor Manager work within the Information Space. It is through the use of information measures and evaluations of goals that

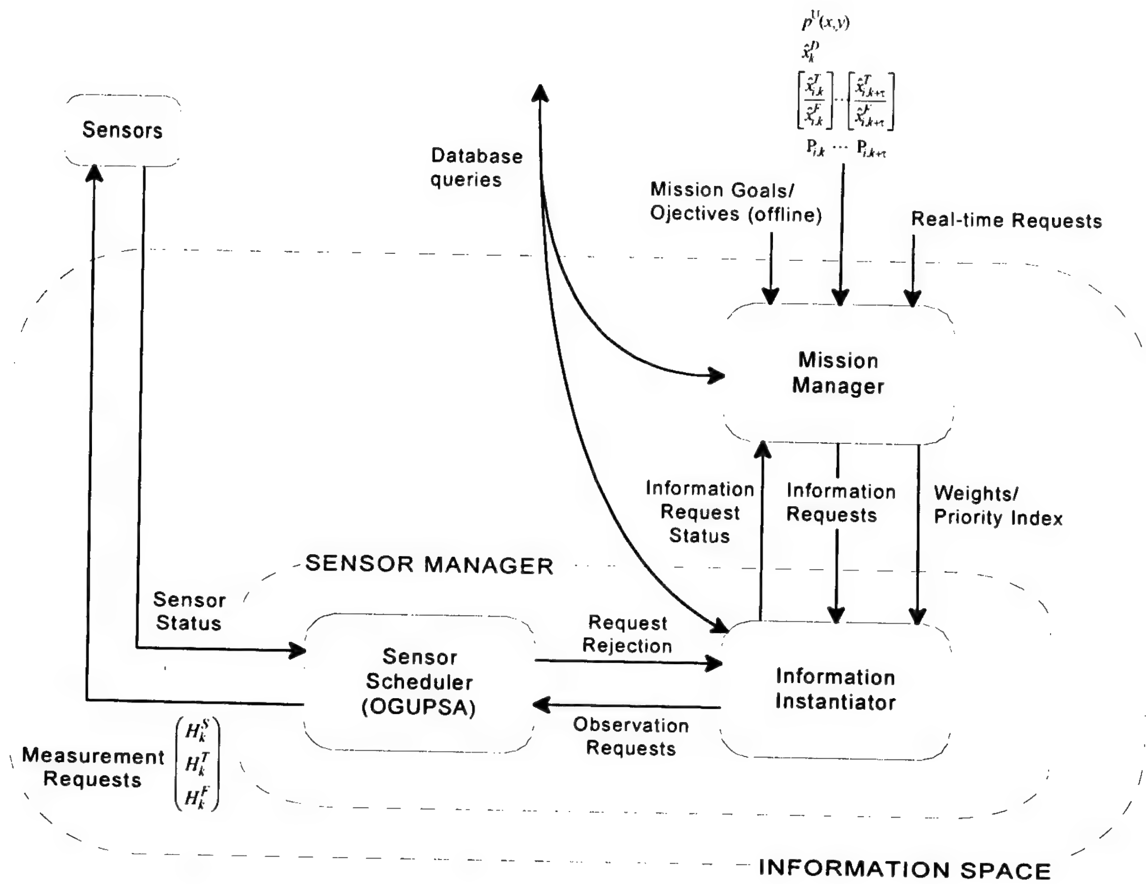


Figure 2-5: Information Space Block Diagram

the Mission Manager computes information requests and the Sensor Manager converts these information request to actual sensor measurements through the intermediate step of observations requests to the information instantiator. Of particular interest is the role of the sensor manager in that it subsumes two, essentially orthogonal tasks, information acquisition management and sensor scheduling.

Previous approaches have treated the sensor management problem as a single optimization task with a performance measure as a weighted sum of diverse measures. Since the goals of the

information-to-observation instantiation are fundamentally those of mapping the observation functions to individual sensors or pseudo-sensors, the two processes can be partitioned into two distinct processes. These two processes can be individually locally optimized (possibly globally suboptimal) based on separate performance measures predicated on appropriate, yet necessarily imperfect, models of the other processes which they subsume. That is, the information manager instantiates requests for information into the specific type of observation which will satisfy that requirement without regard to the particular sensor which will be used to perform the observation. In this manner, it can maximize the flow of information from the world into the information space representation of the world without investigating all options. That is, it makes an optimal decision based on an imperfect and incomplete model of the actual sensors, but in doing so, it reduces the optimization to one which is manageable and calculable in real time.

The sensor scheduler, on the other hand, does not need to know how the measurements are going to satisfy some higher requirement for information. It only needs to concern itself with the optimal packing of these measurements into the time allotted as well as distributing the measurement tasks among the available sensors while simultaneously keeping the load balanced and assuring that all sensors are utilized to their maximum capability.

For example, the Information Instantiator does not care whether an ESM, FLIR, or RADAR is used to obtain a bearing that it needs to improve the estimate of a target's state. It is only concerned with the fact that it needs an observation of a particular type and accuracy level with which to compute the information to satisfy a higher level request. That is, the Information Instantiator only needs to have some bound on the information rate which can be achieved with

the sensor suite, without regard for the specific sensors. In the ideal case, there is some feedback from the sensor scheduler to the Information Instantiator reflecting its real-time capabilities as they degrade or additional sensors come on-line. Likewise, the sensor scheduler is not concerned with the reason for the observations, but is only concerned with the resources that it has available to fulfill the observation requests.

Another way to look at this is that the information manager does not perform micro-management, but assumes that within some bounds, the sensor scheduler can satisfy most of its measurement needs. Those that it can't satisfy are returned to be reprioritized or discarded. It further assumes that the information manager has approximate models of the sensors from which it can obtain measurements, but has no particular interest in which specific sensor the sensor scheduler uses.

2.5 Applying Partially Ordered Sets to Sensor Management

Difficulty arises when trying to prioritize or determine the weight for each management function in order to perform the requisite trade-offs. The use of POSETs and lattices allows one to superimpose a method of apportioning weights to the mission goals that a sensor management system supports. This method is unique to this research and represents a quantum step forward for the integration of "soft" goals with hard limitations.

2.5.1 Partially Ordered Sets and Lattices Theory

The theory of orderings or ordering relations plays an important role throughout mathematics [74] and it is this method that will be effectively applied to the development of a comprehensive

sensor management system. As a preliminary, several definitions are useful. When talking of ordering relations, it's convenient to read the symbol "<" as "is included in" rather than the more usual "is numerically less than". A partially ordered set or POSET is defined as "...a set which has a relation $x < y$, or 'x precedes y', defined for some members or x and y satisfying the conditions: (1) If $x < y$ then $y < x$ is false and x and y are not the same element. (2) If $x < y$ and $y < z$, the $x < z$ [72]." More specifically, a POSET is based on an ordered pair $(X, \leq)$, where X is a set and $\leq$ is an operation or dyadic inclusion relation over X that must satisfy the three requirements of reflexivity, asymmetry, and transitivity [74], [75], [76]. These properties are defined as:

- For all $x \in X$, $x \leq x$ (Reflexive)
- For all $x, y \in X$, if $x \leq y$ and $y \leq x$, then $x = y$ (Asymmetric)
- For all $x, y, z \in X$, if $x \leq y$ and $y \leq z$, then $x \leq z$ (Transitive)

If all of the orderings are not specified, then the ordering relationship is called a partial ordering. If the POSET is further restricted such that for any two elements in the POSET have both a greatest lower bound (glb) and a least upper bound (lub), then the elements form a lattice. Usually a lattice represents the relationship among the elements of a set. A common example of this is the Hasse diagram.

Two examples are presented here to provide an intuitive understanding of POSETs and lattices for those unfamiliar with the concept. While POSETs occur throughout mathematics and are used extensively for machine minimization in Sequential Machine Theory, most examples are based on algebras or sets. Two examples shown below use a convenient illustration of a POSET called the order diagram or Hasse diagram. The first example is an algebraic one

Table 2-3: Integer Inclusion Ordering Relations

Element of X	"integer divisors of"
100	100, 50, 25, 20, 10, 5, 4, 2, 1
50	50, 25, 10, 5, 2, 1
25	25, 5, 1
20	20, 10, 5, 4, 2, 1
10	10, 5, 2, 1
5	5, 1
4	4, 2, 1
2	2, 1
1	1

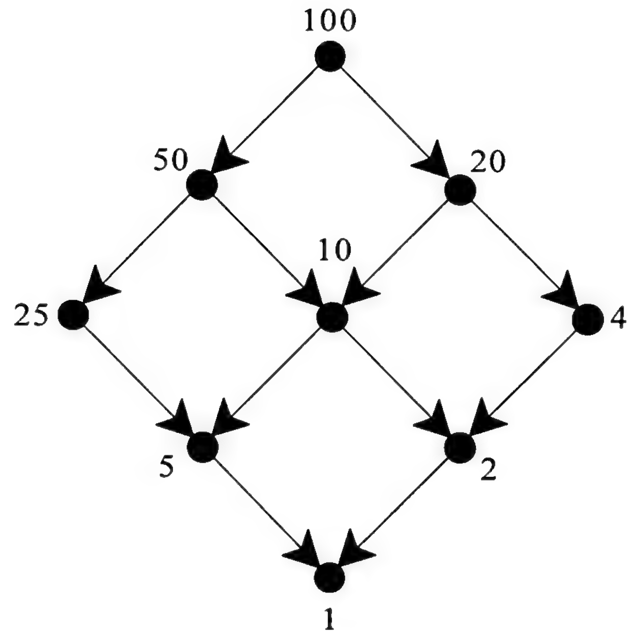


Figure 2-6 Lattice For Integer Division Ordering Relation

based on the inclusion relationship "is an integer divisor of." The relationship is defined as $R{:}x \leq y$ with $\leq$ defined as integer divisor and the set $X=\{1, 2, 4, 5, 10, 20, 25, 50, 100\}$. The resulting inclusion relationship for X is shown in Table 2-3 and the accompanying Hasse diagram (or lattice) of the POSET is shown in Figure 2-6. The second example is based on the relationship $R{:}x \leq y$ where $\leq$ is defined as "is a subset of" and the set is X={ {abcdef}, {{ab},{ef}}, {{ad},{bcef}}, {ab}, {ef}, {a,b,c,d,e,f} }. The lattice for this POSET based on the subset ordering relation is shown in Figure 2-7.

2.5.2 Computing Weights Using POSETs

As discussed earlier, expected information gained has turned out to be a necessary but not sufficient condition to perform the necessary task trade-offs required for complete sensor

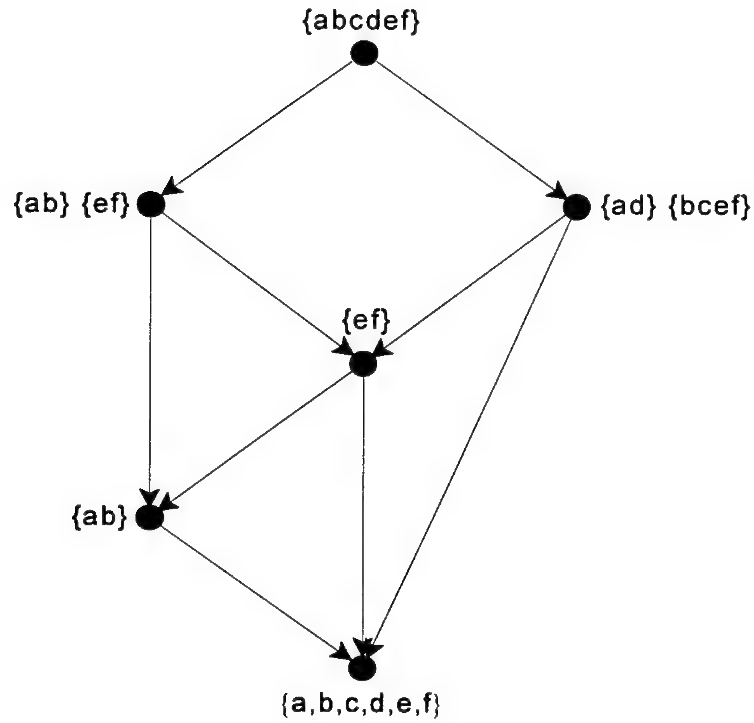
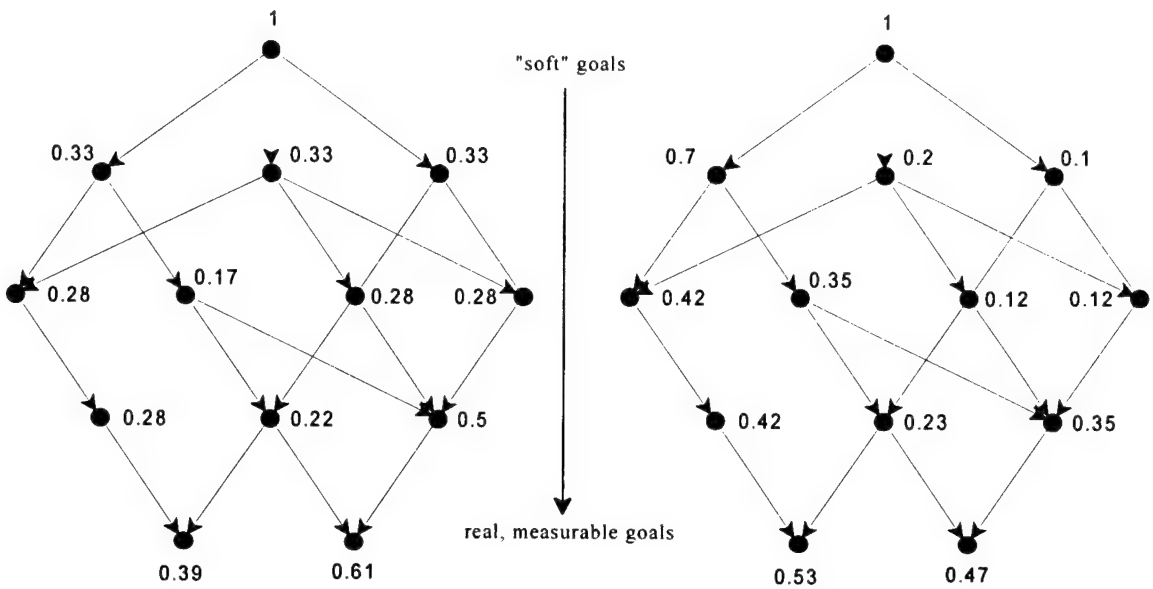


Figure 2-7: Lattice of Subset Ordering Relation

management. That is, if one uses only information gained as a measure with which to perform sensor management trade-offs, it does not take into account the multiplicity of competing mission goals which must also be considered. By defining an ordering relation among the mission goals, the theory of order relations, or more specifically partially ordered sets, can be used to construct a set of goals into a lattice and superimpose on this a method of apportioning relative values among the goals. The values can be determined by starting with the top node of a POSET having a weight of 1. In the absence of any overriding preferences such as changing mission requirements, the value of a goal is uniformly distributed among the arcs leaving that node. The value for each node is then computed by summing the values of all the incoming arcs. An example of a lattice with uniformly apportioned values and 13 nodes is shown in Figure 2-8a. The bottom two nodes, which represent actions which can be performed, represent the lowest



a) Uniformly apportioned values b) User preference apportioned values

Figure 2-8: Lattice with Values Apportioned Uniformly Versus User Preference

level goals and their values are 0.39 and 0.61, respectively. These values can then be used for deciding how frequently to perform which actions or what the relative priorities of the individual actions should be.

In the case of changing user preferences, the values can be distributed among the outgoing arcs according to these preferences rather than the previous uniform distribution. The calculations of the revised values of the lower nodes then is straightforward as described above. For example, if the weights of the three nodes in the second layer are changed to 0.7, 0.2 and 0.1, respectively, the weight of the bottom two nodes change to 0.53 and 0.47, respectively, as shown in Figure 2-8b.

2.5.3 Two Real-World Goal-Lattices

While most sensor management research has been oriented toward military applications, the use of POSETs and lattices can easily be applied to both civilian and military situations to perform and optimize trade-offs among sensor management tasks. The first step in using POSETs is to identify the goals of any given mission. The second step is to define the ordering relation which allows one to build the POSET and associated lattice. The last step is to assign and compute the values for the goals that the sensor manager must trade-off.

A useful civilian example where POSETs and lattices can be applied to is the National Aeronautics and Space Administration (NASA) mission. The NASA's Strategic Plan (dated May 1994) identifies three major mission areas -- scientific research, space exploration, and technology development and transfer. More specifically, NASA [77] lists them as:

- "To explore, use and enable the development of space for human enterprise"
- "Advance and communicate scientific knowledge and understanding of the Earth, the solar system, and the universe, and use the environment of space for research"
- "Research, develop, verify, and transfer advanced aeronautics, space, and related technologies"

Several sub-goals, both from NASA and added by the author, have been identified along with how they relate to the above three mission areas. A complete list of these NASA goals is included in Appendix 1.

An example of a military application is the multiple United States Air Force's (USAF) missions. Several mission areas are defined in the Joint Chief of Staff Publications (JCS Pub 1

and Pub 3) and Air Force doctrinal manuals -- AFM1-1, *Basic Aerospace Doctrine of the United States Air Force* and AFM 1-10, *Combat Support Doctrine* (currently being rewritten as Air Force Doctrine Documents) -- that define and explain Air Force doctrine. These publications and manuals outline six separate mission areas which include Offensive Counterair (OCA), Defensive Counterair (DCA), Air Interdiction (AI), Battlefield Air Interdiction (BAI), Close Air Support (CAS), and Suppression of Enemy Air Defenses (SEAD). Specific goals within each mission area are further described in USAF's Air Command and Staff College course material [78]. These goals are presented in Appendix 2.

2.5.4 Ordering the Goals

Once the goals have been identified (the set, G), as in these 2 examples, the next step is to define an ordering relation (≤) on them which allows one to build a POSET (G, ≤). The ordering relation used in this research is a precedence ordering that simply states that a subordinate goal "is required to accomplish" in order for a goal to be satisfied. Using this ordering relation, a lattice of the POSET based on the NASA mission statement and goals is shown in Figure 2-9 (Note: not all of the subgoals could be identified so the lattice is incomplete leading to the unusual structure of the lattice). The lower portion of the diagram comprises goals for an assumed, but likely, fully autonomous, unmanned Mars explorer with significantly more capabilities than the recently used Sojourner Mars rover. The unordered subgoals of Space Exploration, Scientific Research, and Technology and Transfer are equally weighted with a value of 1/3. The bottom four goals which are real, measurable actions in Figure 2-9 represent the contributing value of the goals of the Mars explorer to the NASA mission. These goals and their weights are:

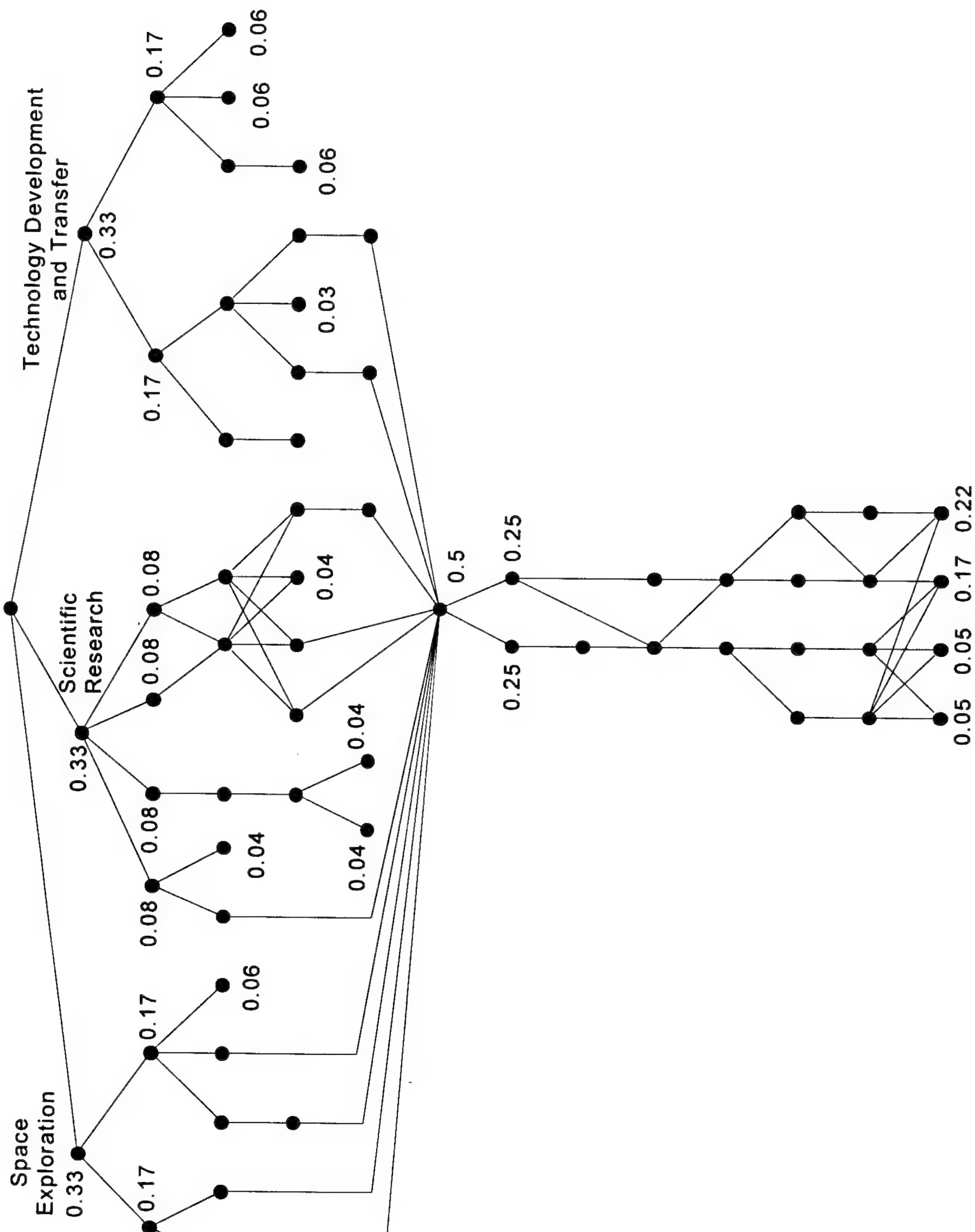


Figure 2-9: NASA Mission Lattice. Details of individual goals are in Appendix 1.

- to analyze the atmosphere, 0.05
- to analyze sample, 0.05
- to search for obstacles, 0.17
- to track obstacles, 0.22

From this, one can see that tracking obstacles contributes more to the NASA mission than analyzing the atmosphere. Therefore, if a decision must be made on whether to do one or the other, tracking should be done first with greater frequency or with a higher rate of occurrence. Also, if there are multiple opportunities then tracking should be done in the ratio of $0.22/0.5$.

The lattice for the POSET based on the USAF goals is shown in Figure 2-10. The six mission areas and their associated weights are annotated in the figure. The bottom three goals and their weights are

- to track detected targets, 0.21
- to id detected targets, 0.22
- to search for targets, 0.57

As previously stated, one of the major advantages of using POSETs with the superimposed value apportionment is that it is a new method that results in a quantitative, and traceable measure of importance that a sensor manager can use to perform and optimize trade-off among search, track, and identify tasks. Another advantage is that the weights can vary as a function of time or state. During any given mission, different goals are preferred over others and these preferences can change during different phases of a particular mission in response to a nonstationary environment. These preference can be set *a priori* and/or in real-time.

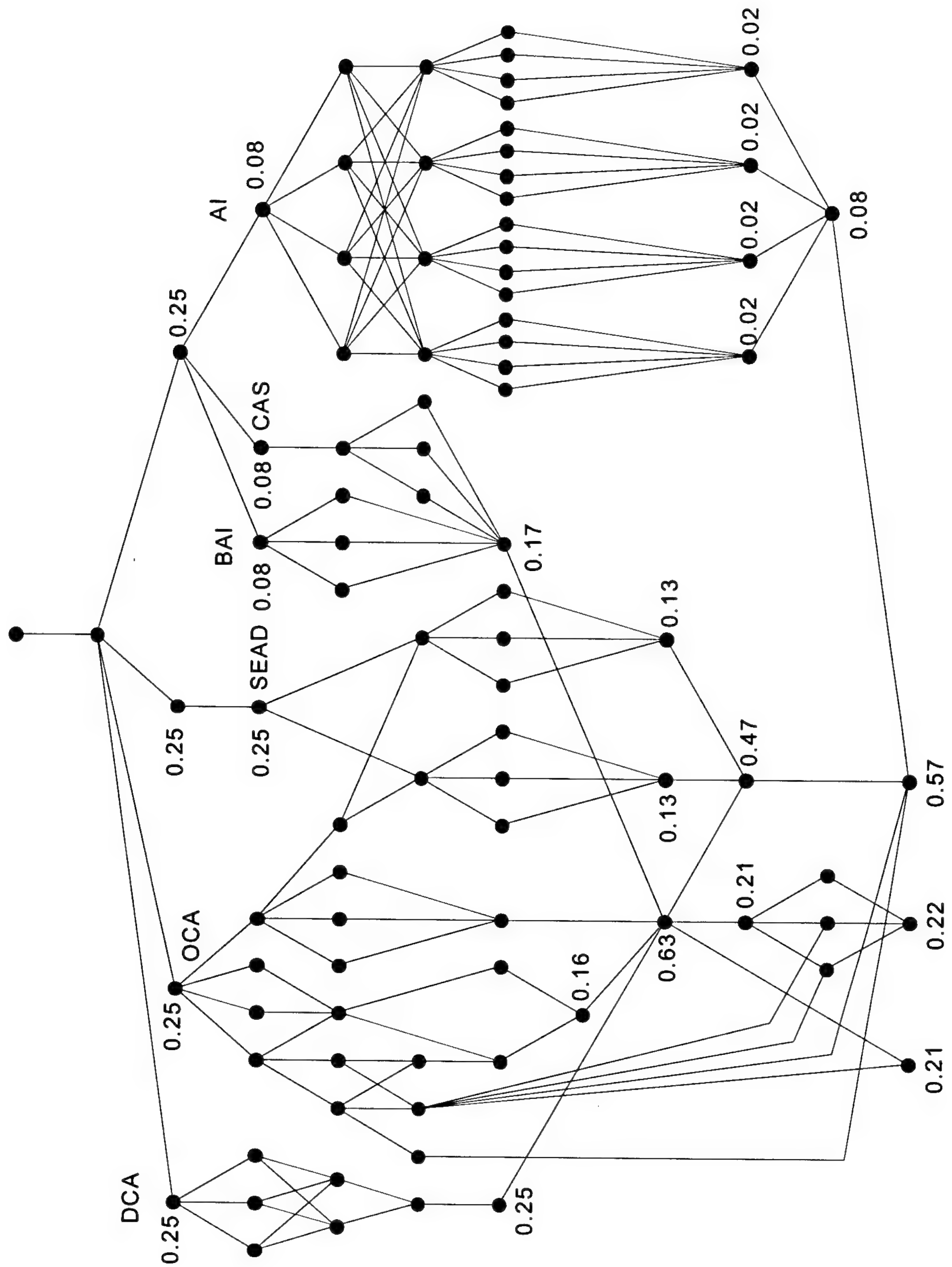


Figure 2-10: USAF Mission Lattice. Details of individual goals are listed in Appendix 2.

Preplanning can establish weights for specific phases of a mission. In the real-time case, a supervisor – either human, automated (*i.e.* the Mission Manager presented earlier), or both – can change the preferences during a mission based on changes in information produced by data fusion. The use of information theory and ordering relations are demonstrated in a simulation model with the results of the simulation runs presented and discussed in Chapter 6.

In summary, the sensor manager is concerned with the detailed scheduling of measurements by the various heterogeneous sensors. It does not concern itself with the particular reason for the measurement, but only with the fact that it has had a request to obtain a measurement of a target. The II determines what functions are required based on the type of request passed to it from the mission manager and the temporal and accuracy constraints of that request. These functions are then converted into tasks and passed along with task deadlines and priorities to the sensor scheduler. The sensor scheduler then optimizes the scheduling of tasks to specific sensors. The Sensor Scheduler (OGUPSA) is discussed in more detailed in Chapter 5. Lastly, the terminology used in this research is that the Mission Manager issues *information requests*, the Information Instantiator issues *observation functions*, and the sensor scheduler issues *sensor actions*.

2.5.5 Goal Lattice Properties

While an initial impression of the goal lattice is that it is nothing more than a graphical belief model, this perception is incorrect. Even though both methods share several similarities, the major difference is in what the methods represent. Graphical belief modeling represents uncertainty by providing a method to build and manipulate risk assessment models [79]. This uncertainty is represented with either probabilities (Bayesian approach) or belief functions

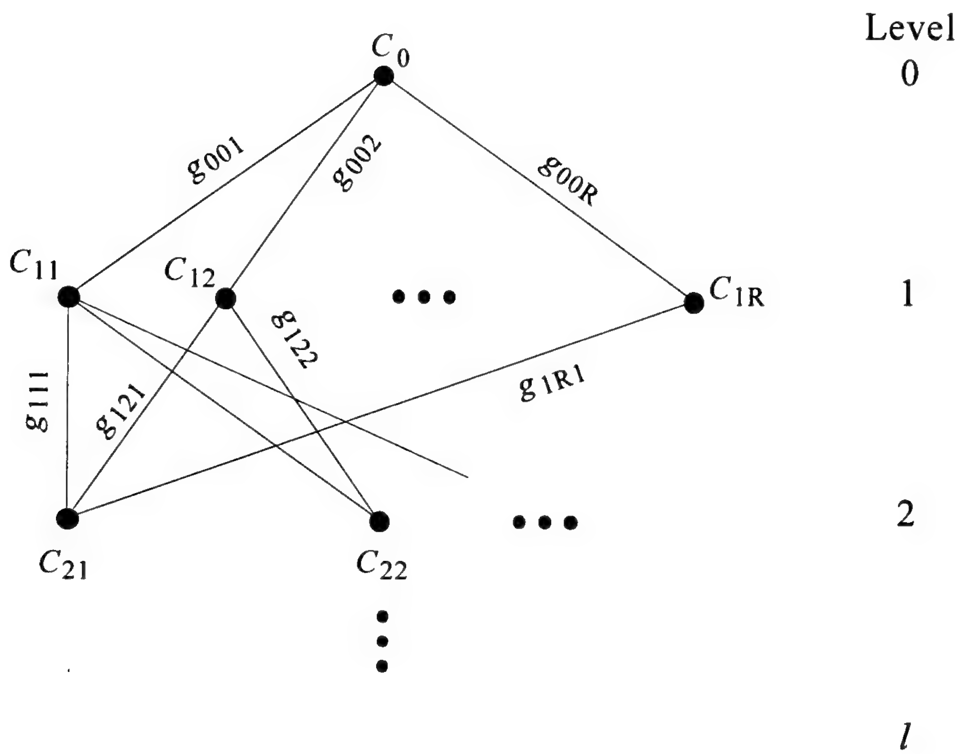


Figure 2-11: Generalized Form of a Goal Lattice

(Dempster-Shafer theory of evidence). The goal lattice was developed as a mathematical method to build and represent user preferences and manipulate them both *a priori*, and more importantly, and in real-time during a mission. The user preferences change during a mission as a function of time (the phase of a mission) or operator input.

A lattice, or Hasse diagram, is used to capture the structure of the sensor management problem. Specifically it uses a mathematical formalization to specify which goals are directly related. While it provides an intuitive description of the problem, it also demonstrates or provides information on how the goal values are influenced when other goal values change. The

goal lattice can be used to translate a complex problem into an easily understood representation and establishes a mechanism for eliciting and documenting an expert's or user's preferences.

As stated in a Section 2.5.2, the value for each node (goal) is computed by summing the values of all the incoming arcs. Using a generalized form of a goal lattice as shown in Figure 2-11, a lattice can be described as having *l* levels with level 0 being the top level, *l-1* middle levels, and level *l* being the bottom level (the level containing the goals whose weights we are attempting to compute). A system of equations can be defined to compute the weights of a particular node at level $i+1$. It is the sum of the products of the incoming arc weight multiplied by the value of the node at level *i* for all nodes that are a parent node. This process continues until the bottom nodes, level *l* nodes, have been defined. For example, the value of the first node at level 2, $c_{2,1}$, is

$$c_{2,1} = g_{1,1,1} * c_{1,1} + g_{1,2,1} * c_{1,2} + \cdots + g_{1,R,1} * c_{1,R} \tag{2-1}$$

where the subscripts of *c* are the level and node within the level (with R nodes in that level). The variable *g* is the user defined arc weights and the subscripts are the level number of the parent node, the node number within that level and the node number in the next level. The sum of all the weights coming from a single node is equal to the value of the node from which they came.

$$c_{i,r} = \sum_{j}^{k} g_{i,r,j} \quad \text{where is } k \text{ the number of arcs leaving } c_{i,r} \tag{2-2}$$

These weights need not be uniformly distributed. Once all of the equations have been defined, the weights of the bottom nodes can be recursively solved such that the weights can be expressed as the sum of value for all of the possible paths from the top node to the bottom node.

The process of solving the system of equations can be extremely tedious and a simpler method is needed. Fortunately, the computation of the goal lattice lends itself to a Linear Algebraic interpretation, [80] and [81], and is easier to visualize. Each layer i can be thought of as a R_i-dimension vector being linear transformed into a R_{i+1}-dimension vector. A matrix, Γ_i, that contains the user specified weights for the arcs leaving the nodes at level i is used to compute the values of the nodes at level i+1. Γ_i can be considered a transformation matrix that transforms the R_i-dimension vector C_i to R_{i+1}-dimension vector C_{i+1} [80]. In matrix form this becomes

$$\mathbf{C}_i = \Gamma_i \mathbf{C}_{i+1} \qquad \textbf{(2-3)}$$

where

$$\mathbf{C}_i' = \begin{bmatrix} \mathbf{C}_{i,1} & \mathbf{C}_{i,2} & \cdots & \mathbf{C}_{i,\mathrm{R}_i} \end{bmatrix}^{\mathrm{T}} \quad \text{a vector of } \mathrm{R}_i \text{ node values for level } i$$

$$\Gamma_i = \begin{bmatrix} g_{i,1,1} & g_{i,2,1} & \cdots & g_{i,\mathrm{R}_i,1} \\ g_{i,1,2} & g_{i,2,2} & \cdots & g_{i,\mathrm{R}_i,2} \\ \vdots & \vdots & \ddots & \vdots \\ g_{i,1,\mathrm{R}_{i+1}} & g_{i,2,\mathrm{R}_{i+1}} & \cdots & g_{i,\mathrm{R}_i,\mathrm{R}_{i+1}} \end{bmatrix}$$

= the transition matrix from level i to level $i+1$ consisting of the arc coefficients. The subscripts of g are

1) from level number i,
2) node number from within level i, and
3) node number within next level

$$\mathbf{C}_{i+1} = \begin{bmatrix} \mathbf{C}_{i+1,1} & \mathbf{C}_{i+1,2} & \cdots & \mathbf{C}_{i+1,\mathrm{R}_{i+1}} \end{bmatrix}^{\mathrm{T}} \quad \text{the vector of } \mathrm{R}_{i+1} \text{ nodes for level } i+1$$

The sum of the columns of Γ_i is the sum of the arcs leaving a node at level i and by the definition of c in (2-2) must sum to 1. If there are a total of R_i nodes in level i and R_{i+1} nodes in the level i+1, then Γ_i will be a $R_i +_1$ by R_i matrix. A transformation matrix can be defined for all of the levels from 1 to l. Once this has been accomplished, each equation of the form in (2-3) can be recursively expanded such as from $\mathbf{C}_1$ and $\mathbf{C}_3$

$$\begin{aligned} \mathbf{C}_2 &= \Gamma_1 \mathbf{C}_l \\ \mathbf{C}_3 &= \Gamma_2 \mathbf{C}_2 \\ &= \Gamma_2 \Gamma_1 \mathbf{C}_l \end{aligned} \qquad (2\text{-}4)$$

The result is that $\Gamma_2\Gamma_1$ is the product of two linear transformation matrices which transforms the vector $\mathbf{C}_1$ into $\mathbf{C}_3$. This new matrix $\Gamma = \Gamma_2 * \Gamma_1$ is itself a linear transformation matrix, [81] and retains the desirable property that the sum of the columns equal 1. Continuing the process from $\mathbf{C}_1$ to $\mathbf{C}_l$ result in the value vector at level l being

$$\mathbf{C}_l = \Gamma_{l-1} * \Gamma_{l-2} * \cdots * \Gamma_1 \mathbf{C}_1 \qquad (2\text{-}5)$$

$$\begin{bmatrix} c_{l,1} \\ c_{l,2} \\ \vdots \\ c_{l,\mathrm{R}l} \end{bmatrix} = \Gamma_{l-1} * \Gamma_{l-2} * \cdots * \Gamma_1 \begin{bmatrix} c_{1,1} \\ c_{1,2} \\ \vdots \\ c_{1,\mathrm{R}1} \end{bmatrix}$$

which is the linear transformation from $\mathbf{C}_1$ to $\mathbf{C}_l$.

A numerical example based on Figure 2-12 is presented to demonstrate the above process. First, the systems of equations are developed and they are

$$\begin{aligned} c_{1,1} &= 0.55 \\ c_{1,2} &= 0.15 \\ c_{1,3} &= 0.2 \\ c_{1,4} &= 0.06 \\ c_{1,5} &= 0.04 \\ c_{2,1} &= 0.5c_{1,1} + 0.6c_{1,2} + 0.7c_{1,4} \\ c_{2,2} &= 0.4c_{1,2} + 1c_{1,3} + 0.3c_{1,4} \\ c_{2,3} &= 0.5c_{1,1} + 1c_{1,5} \\ c_{3,1} &= 0.75c_{2,1} + 0.38c_{2,2} \\ c_{3,2} &= 0.25c_{2,1} + 0.62c_{2,2} + 1c_{2,3} \end{aligned}$$

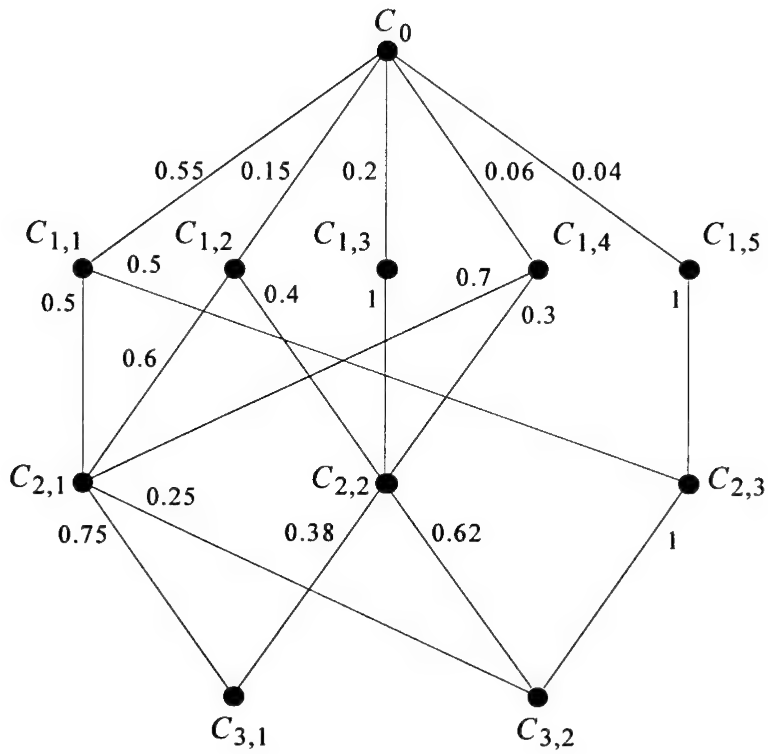


Figure 2-12: Goal Lattice Properties Example

Then solving for $c_{3,1}$ and $c_{3,2}$ yields

$$\begin{aligned}
c_{3,1} &= 0.75\left(0.5c_{1,1} + 0.6c_{1,2} + 0.7c_{1,4}\right) + 0.38\left(0.4c_{1,2} + 1c_{1,3} + 0.3c_{1,4}\right) \\
&= 0.75\left(0.5\right)c_{1,1} + 0.75\left(0.6\right)c_{1,2} + 0.75\left(0.7\right)c_{1,4} + 0.38\left(0.4\right)c_{1,2} \\
&\quad + 0.38\left(1\right)c_{1,3} + 0.38\left(0.3\right)c_{1,4} \\
&= 0.2062 + 0.0675 + 0.0315 + 0.0228 + 0.076 + 0.0068 \\
&= 0.4109 \\
c_{3,2} &= 0.25\left(0.5c_{1,1} + 0.6c_{1,2} + 0.7c_{1,4}\right) + 0.62\left(0.4c_{1,2} + 1c_{1,3} + 0.3c_{1,4}\right) \\
&\quad + 1\left(0.5c_{1,1} + 1c_{1,5}\right) \\
&= 0.25\left(0.5\right)c_{1,1} + 0.25\left(0.6\right)c_{1,2} + 0.25\left(0.7\right)c_{1,4} + 0.62\left(0.4\right)c_{1,2} \\
&\quad + 0.62\left(1\right)c_{1,3} + 0.62\left(0.3\right)c_{1,4} + 1\left(0.5\right)c_{1,1} + 1\left(1\right)c_{1,5} \\
&\quad 0.0688 + 0.0225 + 0.0105 + 0.0372 + 0.124 + 0.0112 + 0.275 + 0.04 \\
&\quad 0.5891
\end{aligned}$$

Thus there are 6 paths from the top node to the bottom node $c_{3,1}$ and are listed below along with their associated path value

$c_0 \rightarrow c_{1,1} \rightarrow c_{2,1} \rightarrow c_{3,1}$ (0.2062),
$c_0 \rightarrow c_{1,2} \rightarrow c_{2,1} \rightarrow c_{3,1}$ (0.0675),
$c_0 \rightarrow c_{1,4} \rightarrow c_{2,1} \rightarrow c_{3,1}$ (0.0315),
$c_0 \rightarrow c_{1,2} \rightarrow c_{2,2} \rightarrow c_{3,1}$ (0.0228),
$c_0 \rightarrow c_{1,3} \rightarrow c_{2,2} \rightarrow c_{3,1}$ (0.0076), and
$c_0 \rightarrow c_{1,4} \rightarrow c_{2,2} \rightarrow c_{3,1}$ (0.0068)

From the top node to the other bottom node, $c_{3,2}$ there are 8 paths. The paths along with their associated weight are

$c_0 \rightarrow c_{1,1} \rightarrow c_{2,1} \rightarrow c_{3,2}$ (0.0688),
$c_0 \rightarrow c_{1,2} \rightarrow c_{2,1} \rightarrow c_{3,2}$ (0.0225),
$c_0 \rightarrow c_{1,4} \rightarrow c_{2,1} \rightarrow c_{3,2}$ (0.0105),
$c_0 \rightarrow c_{1,2} \rightarrow c_{2,2} \rightarrow c_{3,2}$ (0.0372),
$c_0 \rightarrow c_{1,3} \rightarrow c_{2,2} \rightarrow c_{3,2}$ (0.124),
$c_0 \rightarrow c_{1,4} \rightarrow c_{2,2} \rightarrow c_{3,2}$ (0.0112),
$c_0 \rightarrow c_{1,1} \rightarrow c_{2,3} \rightarrow c_{3,2}$ (0.275), and
$c_0 \rightarrow c_{1,5} \rightarrow c_{2,3} \rightarrow c_{3,2}$ (0.04),

The computational complexity of determining the weights of the bottoms nodes can be seen to be polynomial since they are just the sum of the product of the segment weights for each path.

Using the linear transformation representation described above, the level 1 vector is

$$\begin{aligned}\mathbf{C}_1 &= \begin{bmatrix} c_{1,1} & c_{1,2} & c_{1,3} & c_{1,4} & c_{1,5} \end{bmatrix}^T \\ &= \begin{bmatrix} 0.05 & 0.15 & 0.2 & 0.06 & 0.04 \end{bmatrix}^T\end{aligned} \qquad \textbf{(2-6)}$$

and the 2 transformation matrices from level 1 to level 2 (Γ_1) and level 2 to level 3 (Γ_2) are

$$\Gamma_1 = \begin{bmatrix} 0.5 & 0.6 & 0 & 0.7 & 0 \\ 0 & 0.4 & 1 & 0.3 & 0 \\ 0.5 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \textbf{(2-7)}$$

$$\Gamma_2 = \begin{bmatrix} 0.75 & 0.38 & 0 \\ 0.25 & 0.62 & 1 \end{bmatrix}$$

The transformation matrices from level 1 to level 3 ($\Gamma_1 * \Gamma_2$) is given by

$$\begin{aligned} \Gamma &= \Gamma_2 * \Gamma_1 \\ &= \begin{bmatrix} 0.375 & 0.602 & 0.38 & 0.639 & 0 \\ 0.625 & 0.398 & 0.62 & 0.361 & 1 \end{bmatrix} \end{aligned} \quad \textbf{(2-8)}$$

Now using (2-6) and (2-8), $\mathbf{C}_3$ can be computed as follows

$$\begin{aligned} \mathbf{C}_3 &= \Gamma * \mathbf{C}_1 \\ &= \begin{bmatrix} 0.375 & 0.602 & 0.38 & 0.639 & 0 \\ 0.625 & 0.398 & 0.62 & 0.361 & 1 \end{bmatrix} * \begin{bmatrix} 0.55 \\ 0.15 \\ 0.2 \\ 0.06 \\ 0.04 \end{bmatrix} \\ &= \begin{bmatrix} 0.4109 \\ 0.5891 \end{bmatrix} \end{aligned} \quad (2\text{-}9)$$

As discussed previously, the sum of the columns for Γ_i sum to 1 as does Γ. It is interesting to note that the elements of Γ take on special significance. Looking at the 1st column of Γ, this is the proportion of the value of node $c_{1,1}$ that goes to support the bottom goals $c_{3,1}$ and $c_{3,2}$ -- 0.375 and 0.625 respectively. Column 2 is the proportion of the value of node $c_{1,2}$ that supports the bottom goals and so on for the rest of the columns. The elements of each row of Γ also have significance - namely that they are the portion of the level 1 nodes that support the bottom node associated with that row.

2.5.5.1 Goal Lattice Sensitivity

Lattices can be described based on their visual appearance - that is whether or not they are symmetrical. This symmetry or asymmetry can then be used to study the sensitivity of the weights of the goal nodes (the bottom nodes in a goal lattice) to changes in user value preferences of the arcs leaving higher level nodes. This sensitivity can be divided into two categories - value sensitivity and structural sensitivity. Value sensitivity deals with how sensitive the goal nodes are to changes in user arc value preferences while structural sensitivity is concerned with how sensitive the goal nodes are to the asymmetry of the goal lattice.

2.5.5.1.1 Value Sensitivity

In order to demonstrate value sensitivity, a 3 layer symmetric lattice with two bottom nodes is used. The top most goal is divided equally among n nodes in the middle level - each arc has weight $1/n$. One of the arc weights to the n nodes is "perturbed" by the differential value δ while the other n-1 arcs are uniformly decreased by $\delta / (n-1)$. The goal lattice is symmetrical in structure by mirroring it about the vertical axis. The measure of asymmetry is " p ", the number of goals from the middle layer which contribute to each of the 2 bottom most goals (A and B). This goal lattice is depicted in Figure 2-13.

Now using the matrix notation described above, the vector of the values for the nodes in level 1 is a column vector of size n and is given by

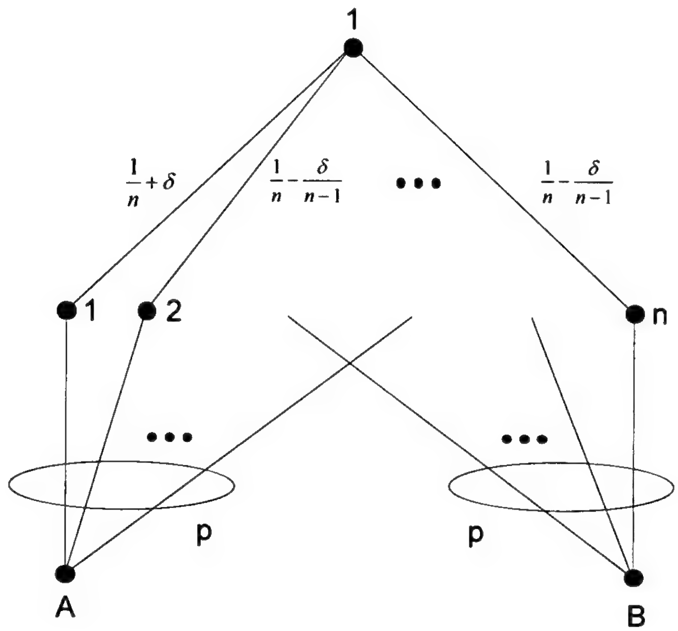


Figure 2-13: Goal Lattice for Sensitivity Example

$$\mathbf{C}_1 = \begin{bmatrix} \frac{1}{n}+\delta \\ \frac{1}{n}-\frac{\delta}{n-1} \\ \vdots \\ \frac{1}{n}-\frac{\delta}{n-1} \end{bmatrix} \tag{2-10}$$

and the value of the lowest nodes, A and B, can be computed as

$$\mathbf{C}_2 = \Gamma\,\mathbf{C}_1 \tag{2-11}$$

$$\begin{bmatrix} \mathrm{A} \\ \mathrm{B} \end{bmatrix} = \left[\begin{array}{c:c:c} \vec{1}_{(p-m \times 1)} & \vec{0.5}_{(m \times 1)} & \vec{0}_{(n-p \times 1)} \\ \vec{0}_{(n-p \times 1)} & \vec{0.5}_{(m \times 1)} & \vec{1}_{(p-m \times 1)} \end{array}\right] \left[\begin{array}{c} \frac{1}{n}+\delta \\ \hdashline \left[\frac{1}{n}-\frac{\delta}{n-1}\right]_{n-1 \times 1} \end{array}\right]$$

where

n is the number of nodes at the middle level

p is the number of nodes from the middle level which contribute to each of the bottom nodes

m is the number of nodes at the middle level shared by each of the bottom nodes

Expanding into separate equations for A and B yields

$$\mathrm{A} = \left(\frac{1}{n} + \delta\right) + (p - m - 1)\left(\frac{1}{n} - \frac{\delta}{n-1}\right) + 0.5m\left(\frac{1}{n} - \frac{\delta}{n-1}\right)$$

$$\mathrm{B} = 0.5m\left(\frac{1}{n} - \frac{\delta}{n-1}\right) + (p - m)\left(\frac{1}{n} - \frac{\delta}{n-1}\right)$$

Then A - B is

$$\begin{aligned}\mathrm{A} - \mathrm{B} &= \left(\frac{1}{n} + \delta\right) + (p - m - 1)\left(\frac{1}{n} - \frac{\delta}{n-1}\right) + 0.5m\left(\frac{1}{n} - \frac{\delta}{n-1}\right) \\ &\quad - 0.5m\left(\frac{1}{n} - \frac{\delta}{n-1}\right) - (p - m)\left(\frac{1}{n} - \frac{\delta}{n-1}\right) \\ &= \left(\frac{1}{n} + \delta\right) - \left(\frac{1}{n} - \frac{\delta}{n-1}\right) \\ &= \delta + \frac{\delta}{n-1} \\ &= \frac{n\delta}{n-1}\end{aligned} \tag{2-12}$$

Taking the partial of A - B with respect to δ yields

$$\begin{aligned}\frac{\partial(\mathrm{A} - \mathrm{B})}{\partial\delta} &= \frac{\partial}{\partial\delta}\left(\frac{n\delta}{n-1}\right) \\ &= \frac{n}{n-1}\end{aligned} \tag{2-13}$$

Taking the partial with respect to n yields

$$\frac{\partial(\mathrm{A} - \mathrm{B})}{\partial n} = \frac{\partial}{\partial n}\left(\frac{n\delta}{n-1}\right) \tag{2-14}$$

$$= \frac{\delta}{(n-1)^2}$$

The sensitivity of the value of the bottom nodes to changes or "perturbations" of the values at the top of the lattice can be described by noticing that as $\delta \rightarrow 0$, A - B = 0 and as $n \rightarrow \infty$, A - B = δ. This simply shows that the smaller the "perturbation" the smaller the effect on the bottom goals. Also the more nodes at the middle level, the smaller the effect of the "perturbation".

2.5.5.1.2 Structural Sensitivity

This process can be repeated for an asymmetric goal lattice in order to examine the sensitivity of the values of the bottom nodes to changes in the structural asymmetry of the goal lattice. The same goal lattice in Figure 2-13 can be used except there is no perturbation and p arcs contribute to bottom node A and q arcs contribute to bottom node B with $p \neq q$ and m arcs in common. The value of A and B can be computed as

$$\mathbf{C}_2 = \Gamma\, \mathbf{C}_1 \tag{2-15}$$

$$\begin{bmatrix} \mathrm{A} \\ \mathrm{B} \end{bmatrix} = \left[\begin{array}{c:c:c} \vec{1}_{(p-m \times 1)} & \vec{0.5}_{(m \times 1)} & \vec{0}_{(n-p \times 1)} \\ \vec{0}_{(n-q \times 1)} & \vec{0.5}_{(m \times 1)} & \vec{1}_{(q-m \times 1)} \end{array}\right] \begin{bmatrix} \dfrac{1}{n} \end{bmatrix}_{n \times 1}$$

Expanding separately for A and B yields

$$\begin{aligned} \mathrm{A} &= (p-m)\frac{1}{n} + 0.5m\frac{1}{n} \\ &= (p-0.5m)\frac{1}{n} \\ \mathrm{B} &= 0.5m\frac{1}{n} + (q-m)\frac{1}{n} \\ &= (q-0.5m)\frac{1}{n} \end{aligned}$$

Then solving for A - B is

$$\begin{aligned} \mathrm{A} - \mathrm{B} &= (p - 0.5m)\frac{1}{n} - (q - 0.5m)\frac{1}{n} \\ &= (p - q)\frac{1}{n} \end{aligned} \tag{2-16}$$

Taking the partial with respect to p - q yields

$$\frac{\partial(\mathrm{A} - \mathrm{B})}{\partial(p - q)} = \frac{1}{n} \tag{2-17}$$

Taking the partial with respect to n yields

$$\frac{\partial(\mathrm{A} - \mathrm{B})}{\partial n} = -(p - q)\frac{1}{n^2} \tag{2-18}$$

The sensitivity of the value of the bottom nodes to the amount of asymmetry, as measured by p - q, can be described as $(p - q) \to 0$, A - B = 0 and as $n \to \infty$, A - B = 0. This shows that the more symmetric the lattice, as measured by p - q, the smaller the effect on the bottom goals. Also the more nodes at the middle level, the smaller the effect of the asymmetry.

While specific examples were used here to examine goal lattice sensitivity, this can be expanded to more general cases. That is, in general, one can measure sensitivity by examining the Jacobian of the transformation matrix.

Chapter 3

Information Theory

3.1 Background

The concept of entropy was first introduced by R. Clausius in 1865 when he was studying heat cycles in phenomenological thermodynamics. Since then the term "entropy" has been appropriated by many fields including statistical mechanics (L. Boltzmann in 1872) communications theory (C. L. Shannon in 1948), probability theory, logic linguistics, abstract analysis and number theory [82]. It is Shannon's measure of information that is of practical interest to sensor management and sensor scheduling. As Skagerstam [82] states, Shannon introduced the concept of information theoretic entropy and information based on the concept of a discrete information source as a discrete random process. Shannon [38] defined the entropy information measure as

$$H \equiv -K \sum_{i=1}^{n} p_i \log p_i \qquad (3\text{-}1)$$

where K is any positive constant and p_i as the probability of the i^{th} outcome of the random event. It is the quantities of the form in (3-1) that Shannon states "...play a central role in information theory as measures of information, choice, and uncertainty. The form of H will be recognized as that of entropy as defined in certain formulations of statistical mechanics... ."

Then using the information entropy defined in (3-1), Shannon defined the information, *I*, as the difference of the entropy for two given probability distributions for the random event.

In a C^3I context, the use of sensors is to decrease our uncertainty about the states of the multiplicity of targets which populate our world. Stated another way, sensors are used to reduce the uncertainty about targets -- such as the location, identification, or intent of all targets in a given area of responsibility, essentially our "world." However, the process of sensing the environment is constrained in that sensors cannot observe all parts of the operating environment simultaneously and still have sufficient gain and selectivity to measure individual targets effectively. A trade-off must be made in searching one area at the expense of others. Sensors have a limited field of view, and by the time a sensor revisits a previously observed area a new target may have appeared or a previously detected target may have maneuvered into a different location. The latter will require the sensor to expend limited resources in order to search a larger area in an attempt to reacquire the target. This is all at the expense of increasing the uncertainty of other search areas, possibly losing track of previously detected targets, or identifying previously detected targets. This spatial-temporal mutual exclusivity of sensors can be considered as a constrained communications channel [61].

A basic assumption is that without sensing the world, its entropy or uncertainty about the world is continually increasing. If allowed to continue without sensing, the world becomes a uniformly distributed space of targets. Because different targets have different dynamics and noise driven processes, there is a differential uncertainty increase among them. It is the purpose

of the sensor management system to discover that differential uncertainty and exploit it to minimize our global uncertainty about the world.

From an information-theoretic viewpoint, the purpose of a sensor is to interact with the operating environment in order to reduce the uncertainty about it. By detecting, localizing, and identifying a target or determining that a target is not present results in an information gain (as measured by a reduction in uncertainty). Information is also gained when a sensor is used to increase the accuracy of the kinematic state of a target that is already being tracked. These information gains or reduction in uncertainty can be broken into 3 components. They are [61]:

- uncertainty of the location of undetected targets, $p^U(x,y)$
- uncertainty with the estimate of a target's kinematic state vector, $\hat{x}_k$
- uncertainty about target identity (from identifying a target as friend or foe to determining target classification to identify a specific target tail or hull number), $\hat{x}_k^F$

Despite the apparent applicability of this information theoretic approach, very few references pertaining to the use of Information Theory for the managing and scheduling of sensors can be found in the literature. They can be categorized into the following areas:

- Kalman filtering
- Target detection / recognition
- Data fusion
- Sensor management

3.2 Information Theory Applied to Kalman Filtering

Several papers apply Information-Theoretic (IT) concepts to general estimation problems. In their paper, Kalata and Priemer [83] derive a minimal-error entropy estimator for linear systems. They base their derivations on mutual information between a random process x resolved by the observations z. The authors show that minimizing the error entropy is equivalent to minimizing the mutual information between the prediction error and the observation. By using the entropy error, the authors derive the optimal discrete linear predictor, filter, and smoother involving additive Gaussian noise disturbances. The result is that the optimal entropy error filtering solution is identical to the optimal means square error (discrete Kalman filter) filtering solution shown in Gelb [84]. Additionally, they show that for non-Gaussian cases, the Kalman filter is a minimax entropy error linear filter.

Tomita, *et al.* [85], apply information theory to only filtering problems. Both discrete time and continuous time filters are presented unlike the previous paper that only looked at discrete time Kalman filters. Specifically, the authors state that "… the necessary and sufficient condition for maximizing the mutual information between a state and the estimate is to minimize the entropy of the estimation error." The authors then proceed to construct the discrete and continuous time Kalman filters using the relationship between maximum mutual information and minimum entropy error. Tomita, *et al.* [86], then extend their information theory approach to derive the optimal filter for a continuous time nonlinear system. The conclusion the authors make is that "… mutual information plays the central role for the estimation problems as well as the coding problems discussed by Shannon [6]."

3.3 Information Theory Applied to Target Detection / Recognition

Hoballah and Varshney [87] look at the detection problem using an entropy based cost function in determining the optimum detection. They show that statistical detection can be viewed as maximizing the amount of information transferred through a channel. The authors also show the relationship of mutual information and receiver operating characteristics (P_D and P_{FA}). They also extend the derivation of the optimum threshold and fusion rules based on mutual information for distributed detection situations.

Clark, *et al.* [88], [89], [90], develop and apply an information theoretic measure to evaluate the performance of forward-looking infrared (FLIR) sensors used for target detection in automatic target recognition (ATR) systems. The FLIR systems under investigation by the authors are used to detect and recognize military vehicles against a low clutter background. With a FLIR, one generally receives a signal return that is expressed in terms of the pixel intensities. The pixel intensities are then used to determine the probability density functions (pdf) of the pixels within a target region and in a background region. The authors use these pdfs in developing their "Information Theory Image Measures" (IT IM) based on the relative entropy of the two distributions (see Soofi [91] for a discussion of relative entropy). In contrast to this approach, the current evaluation process is based on human perception. The IT IM was compared to other measure such as target to background contrast and target versus background entropy based on gray levels of pixel images. The authors conclude that the "… Information Theoretic image measure was found more powerful than Contrast and Entropy for separability of different image regions, resulting in much lower false alarm probability."

Hintz [39] also applies an information measure to automatic target recognition (ATR) - specifically when the ATR is used to aid a trained observer to perform target recognition. The type of system being considered is called a cuer -- measurements of one sensor are used to refine and aim another sensor. The approach used by the author is to measure information in terms of subimages that have meaning to the observer and not the entire scene as was used by Clark, *et al.* Hintz interprets entropy as a measure of uncertainty and thus measures information as the change in entropy with the sign determining if information was gained or lost. The form he uses is:

$$I = H_{\text{before observation}} - H_{\text{after observation}} \qquad \textbf{(3-2)}$$

where he defines H as entropy ($-p \ln p$). The author goes on to define several different types of cuer information and presents several numerical examples to demonstrate the quantities of information available for each type.

A final paper by Turner and Bridgewater [92] discusses the use of an information theoretic approach to surveillance of large areas and the detection of targets. There goal is to maximize the amount of information from each interrogation of the search space by a space-based electronically agile radar. By using information theory, they modify the classical binomial sequential detection. Their process is used to adapt the detection threshold in order to extract the maximum amount of information at each step in the detection process. By dividing the search area into cells and establishing a criteria of maximizing the information gain or uncertainty reduction with each dwell of the radar, they determine which cell to visit next.

3.4 Information Theory Applied to Data Fusion

Either a Bayesian or Dempster-Shafer probabilistic models can be used to address data fusion and data management. Using a Bayesian approach, Manyika and Durrant-Whyte [68] compute the expected utility of taking an action. They demonstrate the use of Fisher information and entropy as a measure of information and use this information metric as the expected utility of data fusion.

Greenway, *et al.* [93] investigate communications management within a decentralized multisensor system where a number of distributed nodes each make local decisions on whether to track or identify a target or to communicate target information to other nodes. The authors compare two communications management algorithms constrained by a maximum transmission bandwidth and available bandwidth. The two algorithms are a round robin approach and an information theoretic approach based on entropy considerations.

Oxenham, *et al.* [94], address measures of information for multi-level data fusion. The authors state that the purpose of data fusion is to increase the information content by fusing multiple sources of uncertain information and that a reduction in uncertainty equates to information measured by Hartley information and Shannon entropy. They use a fuzzy set or fuzzy theory approach to categorize uncertainty into ambiguity and vagueness and then further refine and define several types of uncertainty. They then diverge and discuss a measure of information with respect to Dempster-Shafer theory of evidential reasoning and to fuzzy reasoning. While they define several types of uncertainty and provide examples of how to measure them, it is not clear how it is applied to data fusion.

3.5 Information Theory Applied to Sensor Management and Scheduling

Barker [95] investigated the application of information to search theory. He presents and proves a theorem that states that "...subject to a constraint on total search effort, the allocation of search effort that the maximizes the probability of detection also maximizes the entropy of the posterior search distribution."

Hintz and McVey [37] provide the first article on applying a measure of information to sensor management. Their assumption is that a communication channel is running at its capacity and is unable to handle all of the information that is available -- it is running at its Shannon limit. Extending this concept further, they describe a measurement constrained channel -- that is, several targets are being tracked with the use of a separate Kalman filter for each target. Insufficient sensor resources are available and the available sensors must be scheduled to maintain a specified level of track accuracy. Based on this description , they develop a measure of information using the change of entropy in order to determine how to schedule sensors and process the data. Entropy at a given time is defined as the square root of the norm of the conformal error covariance matrix maintained by the Kalman filter. By computing the change in entropy at each measurement opportunity, they develop a method to sequence measurement through a "constrained" channel. By using entropy as a measure of information, they are able to use this method to maximize the amount of information flow at each available sample interval.

Based on the previous work of Hintz and McVey, Schmaedeke [42] uses information gain as the cost function of a Linear Program to optimize the allocation of multiple sensor to track multiple targets at the next time step. As with Hintz and McVey's approach, Schmaedeke uses

the expected information gain based on extrapolating the Kalman filter error covariance and then calculating the updated covariance matrix after a update.

Kastella [46],[47] proposes another information theoretic measure which he terms as "discrimination gain" which is also know as Kullback-Leibler information. He uses the expected discrimination gain to determine the optimal order for searching a set of discrete detection cells in order to detect and track multiple targets.

3.6 Proposed Information Measures

As stated earlier in this chapter, every opportunity a sensor has to observe the environment equates to a certain amount of information which can be obtained about the state of the environment. A fundamental question is how to use this potential information to manage a suite of sensors while maximizing ones net knowledge about the state of the environment. The search/track/identify decision problem is whether to continue to track or identify a previously detected target and with which sensor to use or whether to search for an, as yet, undetected target.

The approach used in this research to computing the amount of information gained is based on entropy considerations. Using Shannon's entropy, (3-1), as a measure of uncertainty, the change in entropy over time measures the decrease in uncertainty or, synonymously, information gained. In the search versus track versus identify trade-off issue, the amount of information gained from a sensor measurement of the environment versus updating either the kinematic or nonkinematic state estimate for a target can be computed and used to determine which option

provides more information - search, track, or identify. The rationale behind using entropy as an amount of information is that it yields a commensurate measure that affords this comparison.

The approach to computing the information gain is based on mutual information - the change in entropy of the *pdf* before a measurement is taken and after it is taken as in (3-2) where entropy, *H*, is computed based on Shannon's entropy formula. Specifically, entropy is defined as

$$\begin{aligned} H_x &= -\sum_i p(x_i)\log p(x_i) && \text{for the discrete case} \\ &= -\int p(x)\log p(x) && \text{for the continuous case} \end{aligned} \qquad \textbf{(3-3)}$$

where $p(x_i)$ is the probability density (mass) function for the continuous (discrete) distribution. The following sections describe how information gain is computed for target detection (search), tracking, and identification.

3.6.1 Target Search Information

Target locations are maintained probabilistically – that is by maintaining a probability density functions (*pdf*). The first *pdf* is used to represent the probable location of an undetected target and is used to determine where to search next. The assumption is that since the number of undetected targets is unknown, once a target has been detected, there is always another target to be detected. Upon detecting a target, its location is maintained separately by the kinematic state estimation process and not as part of the undetected *pdf*.

Typically, sensor performance characteristics are specified by a particular signal to noise ratio (S/N). The approach used here is to model the sensors in terms of their probability of detection (P_D), probability of false alarm (P_{FA}), and beamwidth. The assumption is that a

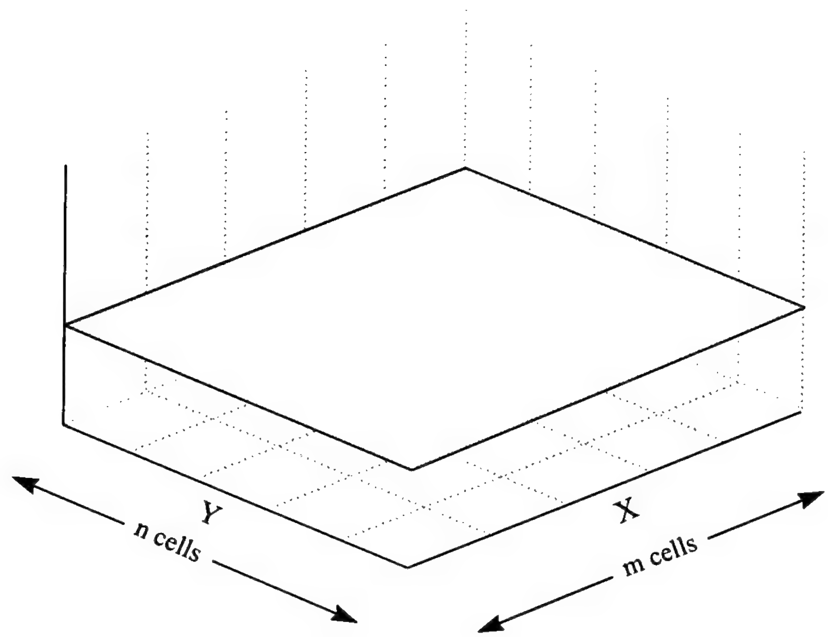


Figure 3-1: Uniform Distributionof Probable Undetected Target Location

particular P_D or P_{FA} can be translated into an equivalent S/N for each sensor and then the S/N can be translated to a particular sensor design. By using this paradigm, it allows any type of sensor to be modeled thus providing the ability to study the effects of different sensors and sensor scheduling schemes.

Given this representation of sensors, the assumptions of the undetected pdf include:

– The search area is represented in Cartesian space (x, y) quantized into m by n cells for a total of $m*n$ cells.

– The initial density function is a function of *a priori* information. In the case of the "in harm's way" situation with no *a priori* information, the initial density function is assumed to be uniform as shown in Figure 3-1. For the uniform case, the location of the undetected

target(s) is unknown so the probability of target being in a cell is $p(\text{tgt in cell } xy) = 1/m*n$ with the distribution function $f(x_i, y_j) = \frac{1}{mn}$.

- $f(x,y)$ is a *pdf* and therefore $\sum_{i=1}^{m} \sum_{j=1}^{n} f(x_i, y_j) = 1$.
- When a sensor performs a measurement, its spatial detection distribution (under the beam) based on beamwidth, P_D, and P_{FA} is converted to an appropriate *pdf* in Cartesian space and use to update the undetected target *pdf*.

Using the above representation, two random variables are hypothesized, **A** and **B**. **B** maps the location of targets in the search area to the integer cells before a measurement is taken. **A** maps these same locations after a measurement is taken. The amount of mutual information between the two random variables can be calculated by the difference in entropy between the **A** and **B** using the discrete case of (3-3).

It is assumed that the MM has access to a two dimension probability density function (pdf) of the operational area which is maintained in real-time by the fusion space. When measurements are made by a sensor, whether they detect a target or not, they influence the pdf of where an undetected target is most likely to be. *A priori* order of battle information can be used to initially skew this pdf to reflect expected target deployment. A target which is detected indicates an area which should not be searched again, although it may be observed in order to convert the target from detection to track and maintain track. A measurement without a detection decreases (to $1\text{-}P_D$) the probability of a target being in that area. After a number of measurements, the probability surface of possible locations of undetected targets becomes quite

convoluted yet does indicate by its peaks the areas which have the highest probability of an undetected target being detected. There is also an ongoing temporal low-pass filtering of the pdf which acts to slowly return the undetected target pdf to a uniform distribution because of the fact that targets could have moved from one unsearched area into an area which has already been searched. Essentially this reflects an increase in world target model entropy as the time since the last measurement increases.

Since the MM has access to this pdf, it uses the goal-lattice derived values to determine when to search as opposed to tracking or identifying. In deciding to request a search, it must pass additional information to the II in order to enable the II to decide what type of observation function to perform. The additional parameters which must be passed include where to search, to what level of certainty to perform the search as measured by $(1 - P_D)$, and a time by which the search must be completed. Note that by only specifying the level of certainty rather than the sensor to use, the II still retains the greatest degree of freedom in determining what type(s) of observation function(s) to request from the sensor manager. Type of function here refers to high- or low- resolution bearing, high- or low- resolution range, Doppler, or some combination thereof. That is, the II takes the general requirements as passed to it from the MM and refines them by determining which of the functions which are available to this sensor suite are capable of supplying the requested information. This approach leaves the actual observation-function-to-sensor-task mapping to OGUPSA.

What can be seen from this first model is a layered approach to optimization in which the MM has an imperfect, coarse model of the target and sensor world with no regard for the actual

manner in which its requirements are going to be satisfied. From this model, search, track, or ID (with the appropriate accuracy and temporal constraints), the MM makes a request which will satisfy its optimization goals which are derived from, or at least quantitatively expressed by, the weighted goal-lattice. It does not bother itself with the implementation details but assumes that there is some mechanism which can be used by the sensor scheduler to meet its needs.

There is, of course, the possibility that the II may not be able to meet the information needs of the MM and hence must reject the request. The MM treats this as another event and, taking into account the rejection along with a reason for the rejection, may chose to make another request with a less stringent information requirement or temporal constraints, or decide that some other information need is more important. This approach leads to a series of parallel local optimization routines which are globally more effective, if not as accurate , as a single, sensor system optimization approach because of the reduced combinatorics of information needs to sensor availability and capability mappings. It is also conceptually more convenient to partition the space of possible alternatives along these lines and possibly apply different optimization criteria to the different layers.

3.6.2 Target Tracking Information

One can also compute the amount of information gain attributed to updating the kinematic state estimate of a target in track. Tracking of a target is probably the simplest and purest information transferring processes. As the target moves, this information degrades between observations and must be updated periodically. At periodic intervals, measurements of the target's position are made and an associated position error covariance is calculated. Assuming

that the errors are Normally distributed and that as time passes since the previous state estimate update, the density flattens (variance increases) but remains Normal. Accordingly, the variance of the position probability density is increasing in the absence of measurements. Said another way, in the interval between measurements, the target's motion increases uncertainty (decreases the amount of information) in its position while the measurement process increases the amount of information about its position.

Track information can be divided into two similar but distinct functions. The first is the transition from detection as a result of a search to tracking a target. The second is the maintenance of a target which is already in track. In the transition-to-track phase, consideration must be made as to how long to wait before taking the next measurement. In the case of a non-Doppler sensor, enough time must elapse between the initial detection and a second measurement in order to get a good estimate of velocity while still maintaining a high probability of detecting the target a second time. If the original detection measurement contains both position and velocity information, then the consideration is one of how soon to make a measurement in order to reduce the error covariance of the state estimate to a level requested by the MM.

Search information has only one temporal constraint, but track information has implied as well as specified temporal constraints associated with the willingness of the MM to tolerate the possibility of a temporary or permanent loss of track. That is, the MM must specify not only the time by which a measurement must be made, but also the maximum error covariance matrix, **P**, which it is willing to accept. The II can use these values which are contained in the request and

combine them with an extrapolation of the error covariance and state obtained from the fusion space to determine how long it can wait for the sensor scheduler to make its measurements and still provide the fusion space with measurements which can be converted into a observation of the accuracy requested by the MM.

Fortunately, the extrapolation of the error covariance can be computed recursively backwards from the requested error covariance matrix ($\mathbf{P}_{req}$) to the $\mathbf{P}^{+}$ of the previous measurement. The net result of this computation is the number of time intervals (or the total elapsed time) between when the previous measurement was made and the time by which the next measurement must be made in order to keep the error covariance below the requested maximum. This requested error covariance may be specified in terms of $\mathbf{P}_{req}$ itself, or some norm defined on the $\mathbf{P}_{req}$.

What this process requires is an appropriate target model that incorporates the maneuver characteristics of the target and a tracking filter state estimator that provides state estimates as well as error measures. One of the most widely used algorithms for such a process is the Kalman filter. As part of the Kalman filter process, and error covariance matrix, **P**, is maintained and propagated. It is this matrix that captures the amount of uncertainty associated with the target's state estimate. With each observation, the error covariance matrix is extrapolated based on the target's motion and then updated resulting in a decrease in uncertainty yielding in a gain in information. The extrapolated covariance matrix, $\mathbf{P}^{-}$, captures the decrease in information due to the target's maneuvers while the updated covariance matrix, $\mathbf{P}^{+}$, captures the increase in

information due to a sensor's measurement. Based on the statistical assumptions of the Kalman filter, $\mathbf{P}^-$ and $\mathbf{P}^+$ can be computed before a measurement is actually made.

Since **P** is a matrix, one must define a norm in order to calculate the entropy such as the determinant of the matrix. Using (3-3) for the *n*-variate case and assuming a normal distribution, the entropy of **P** becomes [96]

$$H_x = \tfrac{n}{2}\log(2\pi e) + \tfrac{1}{2}\log(|\mathbf{P}|) \qquad (3\text{-}4)$$

where $|\mathbf{P}|$ denotes the determinate of the covariance matrix. Defining the information gain between the *a priori* and *a posteriori* entropies as in (3-2), the information gain for the n-variate normal distribution results in

$$\begin{aligned} I &= \tfrac{n}{2}\log(2\pi e) + \tfrac{1}{2}\log(|\mathbf{P}_b|) - (\tfrac{n}{2}\log(2\pi e) + \tfrac{1}{2}\log(|\mathbf{P}_a|)) \\ &= \tfrac{1}{2}\log\left(\frac{|\mathbf{P}_b|}{|\mathbf{P}_a|}\right) \end{aligned} \qquad (3\text{-}5)$$

where $\mathbf{P}_b$ and $\mathbf{P}_a$ are the covariance matrix of the errors before and after a measurement, respectively. This results in the amount of information gained due to the change in the uncertainty about the state of the target.

This measure can be extended to the case of multiple targets and multiple sensors. Since there is no measured entropy change for a target which is not observed, the information gained is due only to the observed target. Since each target in track has its own error covariance matrix, the optimal choice of which target to measure is the one that yields the most information. The assumption is that the global information gain can be maximized by choosing the greatest information gain at each opportunity without regard to future measurements. Each sensor has

different characteristics that include measurement noise. This is accounted for in the propagation of **P**.

There are at least two ways to determine the maximum time between track updates such that an information criterion from the MM is met. The first is to specify a maximum level of uncertainty or uncertainty threshold (as measured by entropy) which is not to be exceeded. The mission manager specifies the uncertainty threshold and the II computes the time when that entropy threshold will be exceeded based on an approximation to the extrapolation of the current error covariance matrix **P** . Using the error covariance extrapolation equation, an information rate (or information rate propagation function if extrapolation of $\mathbf{P}_k^-$ is not linear), or an approximation of this process, can be used to compute the time at which the error covariance matrix will exceed the desired uncertainty.

Given the threshold specified by the mission manager and the error covariance matrix is extrapolated using (3-7), with the entropy computed to determine an information rate using

$$\text{Info threshold} = \text{Info rate} * n \qquad \textbf{(3-6)}$$
$$n = \frac{\text{Info threshold}}{\text{Info rate}}$$

A second approach is also based on the desired level of uncertainty specified by the mission manager, but assumes a constant update (measurement) interval and calculates the actual number of update intervals, *n*, to skip before taking the next measurement. The net effect is the same as the entropy-based approach, however this is an exact approach which may have a closed form solution, and once again shows that this is an II problem which can be solved in different ways

and not a mission manager problem. It is assumed that the MM passes to the II the maximum error covariance that it is willing to accept and which will allow the MM to meet its goals. What is desired is the time at which to make an observation of the target in track in order to produce a $\mathbf{P}_k^+$ which does not exceed this constraint. The problem is how to compute or approximate n, the number of uniform update intervals which are to be skipped while allowing the $\mathbf{P}_k^-$ to propagate and grow. The following shows the development of the equation which must be solved for n.

Given the error covariance extrapolation equation [84]

$$\mathbf{P}_k^- = \Phi_{k-1}\, \mathbf{P}_{k-1}^+\, \Phi_{k-1}{}^T + \mathbf{Q}_{k-1} \tag{3-7}$$

and the error covariance update equation

$$\mathbf{P}_k^+ = \left[\mathbf{I} - \mathbf{K}_k \mathbf{H}_k\right] \mathbf{P}_k^- \tag{3-8}$$

where

$$\mathbf{K}_k = \mathbf{P}_k^-\, \mathbf{H}_k^T \left[\, \mathbf{H}_k\, \mathbf{P}_k^-\, \mathbf{H}_k^T + \mathbf{R}_k \,\right]^{-1} \tag{3-9}$$

if no observation is made at time k, then the observation matrix $\mathbf{H}_k = 0$. Substituting into (3-9) yields

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_k^-\, 0 \left[\, 0\, \mathbf{P}_k^-\, 0 + \mathbf{R}_k \,\right]^{-1} \\ &= 0 \end{aligned} \tag{3-10}$$

Then

$$\begin{aligned} \mathbf{P}_k^+ &= \left[\mathbf{I} - 0\,\right] \mathbf{P}_k^- \\ &= \mathbf{P}_k^- \end{aligned} \tag{3-11}$$

Going back one time step

$$\mathbf{P}_{k-1}^{+} = \mathbf{P}_{k-1}^{-} \tag{3-12}$$

and substituting (3-12) into (3-7) yields

$$\mathbf{P}_{k}^{-} = \Phi_{k-1}\,\mathbf{P}_{k-1}^{-}\,\Phi_{k-1}{}^{T} + \mathbf{Q}_{k-1} \tag{3-13}$$

But

$$\mathbf{P}_{k-1}^{-} = \Phi_{k-2}\,\mathbf{P}_{k-2}^{+}\,\Phi_{k-2}{}^{T} + \mathbf{Q}_{k-2} \tag{3-14}$$

Now substituting (3-14) into (3-13) yields the recursive equation

$$\begin{aligned}\mathbf{P}_{k-1}^{-} &= \Phi_{k-1}\left[\,\Phi_{k-2}\,\mathbf{P}_{k-2}^{+}\,\Phi_{k-2}{}^{T} + \mathbf{Q}_{k-2}\,\right]\Phi_{k-1}{}^{T} + \mathbf{Q}_{k-1} \\ &= \Phi_{k-1}\,\Phi_{k-2}\,\mathbf{P}_{k-2}^{+}\,\Phi_{k-2}{}^{T}\,\Phi_{k-1}{}^{T} + \Phi_{k-1}\,\mathbf{Q}_{k-2}\,\Phi_{k-1}{}^{T} + \mathbf{Q}_{k-1}\end{aligned} \tag{3-15}$$

Continuing backwards to time step *k-n* produces

$$\begin{aligned}\mathbf{P}_{k}^{-} &= \Phi_{k-1}\cdots\Phi_{k-n}\,\mathbf{P}_{k-n}^{+}\,\Phi_{k-n}^{T}\cdots\Phi_{k-1}^{T} + \Phi_{k-1}\cdots\Phi_{k-n-1}\,\mathbf{Q}_{k-n}\,\Phi_{k-n-1}\cdots\Phi_{k-1} + \\ &\quad \Phi_{k-1}\cdots\Phi_{k-n-2}\,\mathbf{Q}_{k-n-1}\,\Phi_{k-n-2}\cdots\Phi_{k-1} + \cdots + \mathbf{Q}_{k-n} \\ &= \left(\prod_{j=1}^{n}\Phi_{k-j}\right)\mathbf{P}_{k-n}^{+}\left(\prod_{j=1}^{n}\Phi_{k-n-j+1}^{T}\right) + \\ &\quad \sum_{j=1}^{n-1}\left(\left(\prod_{m=1}^{n-j}\Phi_{k-m}\right)\mathbf{Q}_{k-n+j-1}\left(\prod_{m=1}^{n-j}\Phi_{k-n+j+m-1}^{T}\right)\right) + \mathbf{Q}_{k-1}\end{aligned} \tag{3-16}$$

If it is assumed that the process is stationary and the transition matrix does not change with time, then $\Phi_{k} = \Phi_{k-1}$ and $\mathbf{Q}_{k} = \mathbf{Q}_{k-1}$ then (3-16) can be simplified to

$$\mathbf{P}_{k}^{-} = \Phi^{n}\,\mathbf{P}_{k-n}^{+}\left(\Phi^{T}\right)^{n} + \sum_{j=1}^{n-1}\left(\Phi^{j}\,\mathbf{Q}\left(\Phi^{T}\right)^{j}\right) + \mathbf{Q} \tag{3-17}$$

allowing $\mathbf{P}_{k}^{-}$ to be expressed in terms of n, $\mathbf{P}_{k-n}^{+}$, and $\mathbf{Q}$.

$\mathbf{P}_{k}^{-}$ can be expressed in terms of $\mathbf{P}_{k}^{+}$, $\mathbf{H}_{k}$, and $\mathbf{R}_{k}$ [84] where

$$\mathbf{P}_{k}^{+\,-1} = \mathbf{P}_{k}^{-\,-1} + \mathbf{H}_{k}^{\mathrm{T}}\,\mathbf{R}_{k}^{-1}\,\mathbf{H}_{k} \tag{3-18}$$

$$\mathbf{P}_k^- = \left(\mathbf{P}_k^{+^{-1}} - \mathbf{H}_k^{\mathrm{T}} \mathbf{R}_k^{-1} \mathbf{H}_k \right)^{-1}$$

Given that the desired $\mathbf{P}_k^+$ is known, the requisite $\mathbf{P}_k^-$ can be computed from (3-18). Using this maximum allowed $\mathbf{P}_k^-$ and the last updated error covariance matrix, $\mathbf{P}_{k-n}^+$, n can be computed from (3-17).

3.6.3 Target Identification Information

There are two aspects to identification information, the first being the obvious reduction in uncertainty about the class of target, the type of target, or the specific (hull-number, side-number) of the target. This is a number which is easily computed from the enumeration of the possible types. A second aspect of identification information is the interaction between ID and target state estimator performance. Most target state estimators are designed based on an assumed target model, the parameters of which change depending on whether the (*e.g.*, airborne) target is a transport, attack aircraft, fighter, or missile. Another confounding aspect of target tracking is the non-stationary statistical behavior of targets, particularly when they are manned and maneuvering. While the model may be the same for these targets with diverse maneuverabilities and non-stationary maneuvering, the ability of the state estimator to maintain track of a target is dependent on the proper choice of filter parameters. In some cases, multiple state estimators with different model parameters are implemented and updated simultaneously and the innovations process is monitored to determine when a maneuver is initiated, indicating that a different state estimator than the current one may be computing the minimum error covariance state estimate.

From the II point of view, it does not care whether the ID is requested to improve tracking performance or to resolve ambiguities about the specific type, class, or hull-number (side-number) of the target. The MM does need to include in its ID request the track number to ID, the time after which the ID would no longer be of value to it, as well as the degree of identification which it needs.

There are a number of goal-oriented reasons for which the MM requests identification information about a target ranging from targeting (Which is the most important target to shoot at?) to improved performance of the target state estimator by providing it with the relative maneuvering class of the target (transport, attack, fighter, missile, *etc.*) so that the target state estimators' assumptions can be improved. Inferential identification, that made from the target track data itself, is done in the data fusion space and requests of this type are the result of specific search or observation requests made by the MM. Direct identification, in which the sensors are asked to reduce the uncertainty about a specific, non-kinematic characteristic of a detection or target in track, do not require the type of calculations previously discussed, but are processed in the II as being mappings from ID information requests to sensor scheduler requests where specific, non-kinematic measurements are scheduled. The II performs a table look-up that determines which type of observations will yield the desired identification. For example, if the MM wants to determine the type of target one could passively use electronic support measures (ESM) to observe the signals emanating from the platform and by consulting the electronic order of battle (EOB) in the fusion space, determine what type of aircraft it is. If a more detailed hull-to-emitter correlation were desired, then some particular ESM characteristics might be used which require a longer observation time. The techniques for identification are numerous and

need not be discussed here other than to indicate that the identification methods, their applicability, and operational constraints can be listed in a table, sometimes with a one-to-many mapping, and these observation options downselected and passed to the sensor scheduler.

Chapter 4

Maneuvering Target Tracking

4.1 Background

Tracking a maneuvering target involves filtering and prediction in order to track the target. "*Filtering* refers to estimating the state vector at the current time, based upon all past measurements. *Prediction* refers to estimating the state at a future time; we shall see that prediction and filtering are closely related [84]." One of the most commonly used technique for target tracking is the discrete Kalman filter developed by Rudolf Kalman. The Kalman filter is *the* optimal linear, unbiased state estimator given its assumptions and is used to filter past measurements and predict where a target will be in the future. This target location prediction is then used to point a sensor in order to track the target. An error covariance matrix is maintained as part of the normal computation process of the Kalman filter. This error covariance matrix can be considered as a measure of uncertainty of the kinematic state (called the state estimate) of the target.

The tracking of maneuvering targets may be complicated by the fact that acceleration may not be directly observable or measurable. Additionally, apparent acceleration can be induced by a variety of sources including human input, autonomous guidance, or atmospheric disturbances. Several approaches to tracking maneuvering targets have been proposed in the literature and can

be divided into two categories both of which assume that the maneuver input command is unknown. One approach is to model the maneuver as a random process. The other approach assumes that the maneuver is not random and that it is either detected or estimated in real time. Both assume a rectilinear model of target track. The random process models generally assume one of two statistical properties, either white noise or an autocorrelated noise. The multiple-model approach is generally used with the white noise model while a zero-mean, exponentially correlated acceleration approach is used with the autocorrelated noise model. The nonrandom approach uses maneuver detection to correct the state estimate or a variable dimension filter to augment the state estimate with an extra state component during a detected maneuver [97].

Another issue to be considered when tracking a maneuvering target is whether to perform the Kalman filter in polar or Cartesian (*x, y*) coordinates. In general, a sensor's measurements are reported in range and bearing (or bearing only in the cases of passive sensors) to the target. If Cartesian coordinate are used, then the range (*r*) and bearing (θ) measurements must be converted through the transformation equations:

$$\begin{aligned} x &= r\cos\theta \\ y &= r\sin\theta \end{aligned} \qquad (4\text{-}1)$$

which results in cross-correlated measurement noise. The resulting covariance matrix can be represented as

$$\mathbf{R}_{xy} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy}^2 \\ \sigma_{xy}^2 & \sigma_y^2 \end{bmatrix} \qquad (4\text{-}2)$$

by using a first order expansion [98], [99], [100] where

$$\sigma_x^2 = \sigma_r^2 \cos^2\theta + r^2\sigma_\theta^2 \sin^2\theta \qquad (4\text{-}3)$$
$$\sigma_y^2 = \sigma_r^2 \sin^2\theta + r^2\sigma_\theta^2 \cos^2\theta$$
$$\sigma_{xy}^2 = \tfrac{1}{2}\sin\theta(\sigma_r^2 - r^2\sigma_\theta^2)$$
$$\sigma_r^2 = \text{range measurement variance}$$
$$\sigma_\theta^2 = \text{bearing measurement variance}$$

In using Cartesian coordinates, the state equation is linear while the corresponding measurement equation is nonlinear. Using polar coordinates, the state equation is nonlinear but the measurement equation is linear [101]. This means that tracking in Cartesian coordinates has the advantage that it allows the use of linear target dynamic models for extrapolation while polar coordinates may lead to more complicated extrapolation. By examining (4-1) and (4-2), using Cartesian coordinates for tracking leads to two major disadvantages. The first is that the measured (or estimated) range must be available while the second is that measurement errors are coupled.

The exponentially correlated acceleration model approach is one of the approaches most widely used to track maneuvering targets. This chapter examines and compare several exponentially correlated acceleration approaches in both polar and Cartesian coordinates for accuracy and computational complexity. They include the Singer model in both polar and Cartesian coordinates, the Sklansky model (not an exponentially correlated acceleration), Helferty's third-order rational approximation of the Singer model, and Bar-Shalom and Fortmann's model.

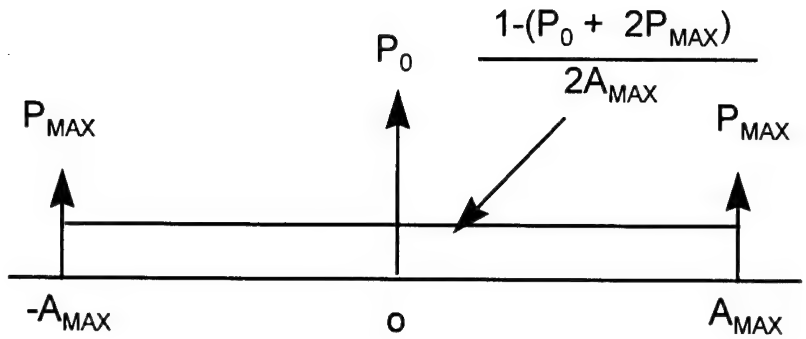


Figure 4-1: Target maneuver probability density function [103]

4.2 Singer Model Using Polar Coordinates

Singer [102], [103], [104] developed a model that incorporates the maneuver capability of a target that is both simple and suitably represents the maneuver characteristics. The Singer model for manned maneuvering targets assumes that a target usually moves at constant velocity and that turns, evasive maneuvers, and accelerations due to atmospheric disturbances can be viewed as perturbations of the constant velocity trajectory. These accelerations are termed target maneuvers and are correlated in time with the previous time or the next time increment. That is to say that if a target is maneuvering at time t, it is likely to be maneuvering at time $t+\tau$ assuming that τ is sufficiently small. Singer [102] states that a lazy turn will give correlated inputs for up to one minute, evasive maneuvers due to radar detection, terrain features, or preprogrammed maneuvers will provide correlated inputs for 10 to 30 seconds, and atmospheric turbulence for only 1 to 2 seconds. Due to this time dependence, the maneuvers are neither additive nor Gaussian. Singer's probability density function for a target's maneuvers are shown in Figure 4-1. A target can [102]:

- Accelerate (maneuver) at its maximum rate, $\pm A_{\max}$ with a probability of $P_{\max}$
- No maneuver with a probability of P_0, or

- Maneuver between - $A_{\max}$ and + $A_{\max}$ according to the uniform distribution shown in Figure 4-1.

In order to use this model in a optimal filter such as a Kalman filter, the maneuver noise needs to be whitened. Singer [103] uses a procedure analogous to the whitening procedure developed by Wiener and Kolmogorov. The whitening processes is done by augmenting the state vector to include the maneuver variables and expressing them recursively in terms of white noise.

The target maneuver model is in polar coordinates and given by the state equation

$$\mathbf{x}_{k+1} = \mathbf{\Phi}\,\mathbf{x}_k + \mathbf{G}\,w_k \qquad (4\text{-}4)$$

where

$$\mathbf{x}_k = \begin{bmatrix} r_k & \dot{r}_k & u_{r,k} & \theta_k & \dot{\theta}_k & u_{\theta,k} \end{bmatrix}^T$$

$$\mathbf{w}_k = \begin{bmatrix} w_{1,k} & w_{2,k} \end{bmatrix}^T$$

$$\Phi = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & \rho & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & \rho \end{bmatrix} \qquad \mathbf{G} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$T =$ sampling period

$\rho =$ correlation coefficient of maneuver

$= e^{\alpha T}$ or $\cong 1 - \alpha T$ if αT is small

$$\mathbf{Q}_\mathrm{k} = E\left[\mathrm{w}_\mathrm{k} w_k^T\right] = \begin{bmatrix} \sigma_{M_1}^2(1-\rho) & 0 \\ 0 & \sigma_{M_2}^2(1-\rho) \end{bmatrix}$$

$$\sigma_{M_1}^2 = \frac{A_{\max}^2 T^2}{3}\left(1 + 4P_{\max} - P_0\right)$$

$$\sigma_{M_2}^2 = \frac{A_{\max}^2 T^2}{3R^2}\left(1 + 4P_{\max} - P_0\right)$$

$R =$ target range

The measurement equation is given by

$$\mathbf{z}_k = \mathbf{H}\,\mathbf{x}_k + \mathbf{v}_k \tag{4-5}$$

where

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\mathbf{R}_k = \begin{bmatrix} \sigma_{r,k}^2 & 0 \\ 0 & \sigma_{\theta,k}^2 \end{bmatrix}$$

The standard filter equations for state estimation extrapolation, error covariance extrapolation, Kalman gain matrix computation, state estimate update, and error covariance updates are then applied. The filter is initialized based on the first two observations with the state estimate given by

$$\hat{\mathbf{x}}_2 = \begin{bmatrix} \mathbf{z}_2(1) & \frac{1}{T}\left(\mathbf{z}_2(1) - \mathbf{z}_1(1)\right) & 0 & \mathbf{z}_2(2) & \frac{1}{T}\left(\mathbf{z}_2(2) - \mathbf{z}_1(2)\right) & 0 \end{bmatrix}^T \tag{4-6}$$

and the nonzero elements of the updated error covariance matrix, $\mathbf{P}_2^+$, defined as

$$\begin{aligned}
P_{11} &= \sigma_r^2 & P_{44} &= \sigma_\theta^2 \\
P_{22} &= \sigma_{M_1}^2 + \left(\frac{2\sigma_r^2}{T^2}\right) & P_{55} &= \sigma_{M_2}^2(1) + \left(\frac{2\sigma_\theta^2}{T^2}\right) \\
P_{33} &= \sigma_{M_1}^2 & P_{66} &= \sigma_{M_2}^2(1) \\
P_{12} &= P_{21} = \frac{\sigma_r^2}{T^2} & P_{45} &= P_{54} = \frac{\sigma_\theta^2}{T^2} \\
P_{23} &= P_{32} = \rho\sigma_{M_1}^2 & P_{56} &= P_{65} = \rho\sigma_{M_2}^2(1)
\end{aligned} \tag{4-7}$$

with σ_{M_1} calculated in (4-4) and

$$\sigma_{M_2}^2(1) = \frac{\sigma_{M_1}^2}{z_1^2(1)} \tag{4-8}$$

4.3 Singer Model using Cartesian Coordinates

A version of the Singer model can be developed for Cartesian coordinates using a constant velocity model with exponentially correlated acceleration. The state equation and measurement model is

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{F}(t)\,\mathbf{x}(t) + \mathbf{G}(t)\,\mathbf{w}_1(t) \\ \mathbf{z}(t) &= \mathbf{H}(t)\,\mathbf{x}(t) + \mathbf{v}(t) \end{aligned} \tag{4-9}$$

where

$$\mathbf{x}(t) = \begin{bmatrix} x(t) & \dot{x}(t) & y(t) & \dot{y}(t) \end{bmatrix}^T$$

$$\mathbf{F}(t) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{G}(t) = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{H}(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

where the process noise is exponentially correlated, assumed to be equally distributed in the x and y directions, and used to model the target acceleration. The measurement noise is normally distributed with zero mean and covariance $\mathbf{R}$ as in (4-2). The process noise can be whitened by augmenting the state vector by appending the necessary state vector components of a linear shaping filter. This results in a linear model driven by white noise. This whitening process is

described in Grewal and Andrews [105] and repeated below. Modeling the correlated noise, $\mathbf{w}_1(t)$, in (4-9) with a shaping filter yields

$$\begin{aligned} \dot{\mathbf{x}}_{SF}(t) &= \mathbf{F}_{SF}(t)\,\mathbf{x}_{SF}(t) + \mathbf{G}_{SF}(t)\,\mathbf{w}_2(t) \\ \mathbf{w}_2(t) &= \mathbf{H}_{SF}(t)\,\mathbf{x}_{SF}(t) \end{aligned} \quad \textbf{(4-10)}$$

where SF denotes the shaping filter and $\mathbf{w}_2(t)$ is a zero mean white Gaussian noise. Using the system model given in (4-9) an augmented state vector is formed and given by

$$\mathbf{X}(t) = \begin{bmatrix} \mathbf{x}(t) & \mathbf{x}_{SF}(t) \end{bmatrix}^T \quad \textbf{(4-11)}$$

Combining (4-9) and (4-10) yields the following augmented system:

$$\begin{aligned} \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}_{SF}(t) \end{bmatrix} &= \begin{bmatrix} \mathbf{F}(t) & \mathbf{G}(t)\,\mathbf{H}_{SF}(t) \\ 0 & \mathbf{F}_{SF}(t) \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_{SF}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{G}_{SF}(t) \end{bmatrix} \mathbf{w}_2(t) \\ \dot{\mathbf{X}}(t) &= \mathbf{F}_T(t)\,\mathbf{X}(t) + \mathbf{G}_T(t)\,\mathbf{w}_2(t) \\ \mathbf{z}(t) &= \begin{bmatrix} \mathbf{H}(t) & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_{SF}(t) \end{bmatrix} + \mathbf{v}(t) \\ &= \mathbf{H}_T(t)\,\mathbf{X}(t) + \mathbf{v}(t) \end{aligned} \quad \textbf{(4-12)}$$

Using Singer's model, the acceleration is uniformly distributed between $-A_{max}$ and A_{max} and the mean number of acceleration changes, α, in a unit time is distributed according to a Poisson process. This results in a first-order Markov process with variance σ^2 and time constant $1/\alpha$. The power spectral density corresponding to this exponential process is

$$\Psi(\omega) = \frac{2\sigma^2\alpha}{\omega^2 + \alpha^2} \quad \textbf{(4-13)}$$

and the system transfer function for the shaping filter is

$$H(s) = \frac{\sigma\sqrt{2\alpha}}{s^2 + \alpha^2} \quad \textbf{(4-14)}$$

The system model for this shaping filter is

$$\dot{\mathbf{x}}_{SF}(t) = \begin{bmatrix} -\alpha \\ -\alpha \end{bmatrix} \mathbf{x}_{SF}(t) + \begin{bmatrix} \sigma\sqrt{2\alpha} \\ \sigma\sqrt{2\alpha} \end{bmatrix} \mathbf{w}_2(t) \tag{4-15}$$

$$\mathbf{w}_2(t) = \begin{bmatrix} 1 \end{bmatrix} \mathbf{x}_{SF}(t)$$

The augmented system then becomes

$$\begin{bmatrix} \dot{x}(t) \\ \ddot{x}(t) \\ \dot{x}_1(t) \\ \dot{y}(t) \\ \ddot{y}(t) \\ \dot{y}_1(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\alpha & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -\alpha \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \\ x_1(t) \\ y(t) \\ \dot{y}(t) \\ y_1(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \sigma\sqrt{2\alpha} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \sigma\sqrt{2\alpha} \end{bmatrix} \mathbf{w}_2(t) \tag{4-16}$$

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \\ x_1(t) \\ y(t) \\ \dot{y}(t) \\ y_1(t) \end{bmatrix} + \mathbf{v}(t)$$

with $\mathbf{w}_2$(t) and $\mathbf{v}$(t) ~ $N(0,1)$.

4.4 Sklansky Model

The Sklansky model is a Cartesian coordinate, constant velocity tracking algorithm that does not model acceleration to generate position and velocity estimates of maneuvering targets [106]. The target motion is described by

$$\begin{aligned} x_{n+1} &= x_n + \mathrm{T}\dot{x}_n + \tfrac{1}{2}\mathrm{T}\ddot{x}_n + \cdots \\ \dot{x}_{n+1} &= \dot{x}_n + \mathrm{T}\ddot{x}_n \end{aligned} \tag{4-17}$$

where

x_n = target position
$\dot{x}_n$ = target velocity
T = time interval between observations
$\ddot{x}_n$ = target acceleration

The state space representation of the Sklansky model is given by

$$\mathbf{x}_{k=1} = \Phi_k \mathbf{x}_k + \mathbf{G}_k \mathbf{a}_k \qquad (4\text{-}18)$$
$$\mathbf{z}_k = \mathbf{H}x_{k=1} + \mathbf{v}_k$$

where

$$\Phi_k = \begin{bmatrix} 1 & \mathrm{T} & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & \mathrm{T} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{x}_k = \begin{bmatrix} x & \dot{x} & y & \dot{y} \end{bmatrix}$$
$$= \begin{bmatrix} x\text{ position} & x\text{ velocity} & y\text{ position} & y\text{ velocity} \end{bmatrix}^{\mathrm{T}}$$

$$\mathbf{G} = \begin{bmatrix} \mathrm{T}^2/2 & 0 \\ \mathrm{T} & 0 \\ 0 & \mathrm{T}^2/2 \\ 0 & \mathrm{T} \end{bmatrix}$$

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$\mathbf{a}_k = \begin{bmatrix} u_x(k) & u_y(k) \end{bmatrix}^{\mathrm{T}}$
= random acceleration in the x and y coordinate respectively
$\mathbf{v}_k$ = scalar random measurement noise with $\mathbf{Q} \sim N(0,1)$

4.5 Helferty Model

Helferty [107] develops a turn-rate model that extends the work of Singer by using a maneuvering target model that combines a constant velocity and a probability distribution on the

target's turn-rate. Helferty assumes that the acceleration is independent in both the x and y coordinates and a uniform distribution on the target's turn rate with the acceleration maneuvers exponentially correlated. This turn-rate model leads to a linear system that is represented with a third-order Markov process instead of the first-order process.

The Helferty model assumes a process noise of constant velocity and the turn-rate uniformly distributed $[-r_{max}, r_{max}]$ with the turn-rate changing α times in a unit interval. The heading angle of the target is also uniformly distributed but on the interval $[-\pi, \pi]$. The autocorrelation function of the target acceleration in the x axis is

$$\begin{aligned}
E\left[a_x(t)a_x(t+\tau)\right] &= E\left[v_t\,\dot\psi(t)\sin\psi(t)\,v_t\,\dot\psi(t+\tau)\sin\psi(t+\tau)\right] \\
&= v_t^2\,E\left[\dot\psi^2\sin\psi(t)\sin\psi(t+\tau)\right]e^{-\alpha|\tau|} \\
&= v_t^2\,E\left[\dot\psi^2\sin\psi(t)\sin\left(\psi(t)+\dot\psi\tau\right)\right]e^{-\alpha|\tau|} \\
&= v_t^2\,E\left[\dot\psi^2\sin\psi(t)\left(\sin\psi(t)\cos\dot\psi\tau+\sin\dot\psi\tau\cos\psi(t)\right)\right]e^{-\alpha|\tau|} \\
&= v_t^2\,E\left[\dot\psi^2\sin^2\psi(t)\cos\dot\psi\tau+\dot\psi^2\sin\psi(t)\sin\dot\psi\tau\cos\psi(t)\right]e^{-\alpha|\tau|} \\
&= v_t^2\,E\left[\dot\psi^2\cos\dot\psi\tau\right]E\left[\sin^2\psi(t)\right]e^{-\alpha|\tau|} \\
&= \frac{v_t^2}{2}E\left[\dot\psi^2\cos\dot\psi\tau\right]e^{-\alpha|\tau|}
\end{aligned} \tag{4-19}$$

The autocorrelation function for the target acceleration in the y axis is

$$\begin{aligned}
E\left[a_y(t)a_y(t+\tau)\right] &= E\left[v_t\,\dot\psi(t)\cos\psi(t)\,v_t\,\dot\psi(t+\tau)\cos\psi(t+\tau)\right] \\
&= v_t^2\,E\left[\dot\psi^2\cos\psi(t)\cos\psi(t+\tau)\right]e^{-\alpha|\tau|} \\
&= v_t^2\,E\left[\dot\psi^2\cos\dot\psi\tau\right]E\left[\cos^2\psi(t)\right]e^{-\alpha|\tau|} \\
&= \frac{v_t^2}{2}E\left[\dot\psi^2\cos\dot\psi\tau\right]e^{-\alpha|\tau|}
\end{aligned} \tag{4-20}$$

and the cross correlation between the x and y axis can be shown to be zero[107].

The power spectral density of the autocorrelation function of (4-19) and (4-20) is nonlinear so Helferty computes and presents a rational approximation for the linear shaping filter for the turn-rate distribution. It is given as

$$H(s) = \frac{b_1 s^2 + b_2 s + b_3}{s^3 + a_1 s^2 + a_2 s + a_3} \tag{4-21}$$

The state equation and measurement model used by Helferty is the same as in (4-9) with

$$\mathbf{x}(t) = \begin{bmatrix} x(t) & \dot{x}(t) & y(t) & \dot{y}(t) \end{bmatrix}^T$$

$$\mathbf{F}(t) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{G}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^T$$

$$\mathbf{H}(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Applying the whitening process described in Section 4.3, the model for the third-order linear shaping filter given in (4-21) for one coordinate is

$$\dot{\mathbf{x}}_{SF}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_3 & -a_2 & -a_1 \end{bmatrix} \begin{bmatrix} x_3(t) \\ x_4(t) \\ x_5(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \mathbf{w}_2(t) \tag{4-22}$$

$$\mathbf{w}_1(t) = \begin{bmatrix} b_3 & b_2 & b_1 \end{bmatrix} \begin{bmatrix} x_3(t) \\ x_4(t) \\ x_5(t) \end{bmatrix}$$

This results in the augmented state and measurement equation

$$\dot{\mathbf{X}}(t)=\begin{bmatrix}0&1&0&0&0&0&0&0&0&0\\0&0&b_3&b_2&b_1&0&0&0&0&0\\0&0&0&1&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0&0\\0&0&-a_3&-a_2&-a_1&0&0&0&0&0\\0&0&0&0&0&0&1&0&0&0\\0&0&0&0&0&0&0&b_3&b_2&b_1\\0&0&0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&0&0&1\\0&0&0&0&0&0&0&-a_3&-a_2&-a_1\end{bmatrix}\mathbf{X}(t)+\begin{bmatrix}0&0\\0&0\\0&0\\0&0\\1&0\\0&0\\0&0\\0&0\\0&0\\0&1\end{bmatrix}\mathbf{w}_2(t) \qquad (4\text{-}23)$$

$$\mathbf{z}(t)=\begin{bmatrix}1&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&1&0&0&0&0\end{bmatrix}\mathbf{X}(t)+\mathbf{v}(t)$$

where

$$\mathbf{X}(t)=\begin{bmatrix}x(t) & \dot{x}(t) & x_3(t) & x_4(t) & x_5(t) & y(t) & \dot{y}(t) & y_3(t) & y_4(t) & y_5(t)\end{bmatrix}$$

and the process noise is normally distributed with zero mean and unit variance.

4.6 Bar-Shalom and Fortmann Model

Another exponentially correlated acceleration model based on the Singer Model is presented by Bar-Shalom and Fortmann [97]. They use a linear shaping filter to augment the Kalman filter. The continuous-time state equation and measurement model is

$$\dot{\mathbf{x}}(t)=\begin{bmatrix}0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&-\alpha&0&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\\0&0&0&0&0&-\alpha\end{bmatrix}\mathbf{x}(t)+\mathbf{w}(t) \qquad (4\text{-}24)$$

$$\mathbf{z}(t)=\begin{bmatrix}1&0&0&0&0&0\\0&0&0&1&0&0\end{bmatrix}\mathbf{x}(t)+\mathbf{v}(t)$$

$$\mathbf{x}(t)=\begin{bmatrix}x(t) & \dot{x}(t) & \ddot{x}(t) & y(t) & \dot{y}(t) & \ddot{y}(t)\end{bmatrix}^T$$

The discrete-time state equation corresponding to (4-24) with sample interval T is

$$\mathbf{x}(k+1) = \mathbf{F}\,\mathbf{x}(k) + \mathbf{w}(k) \tag{4-25}$$

where

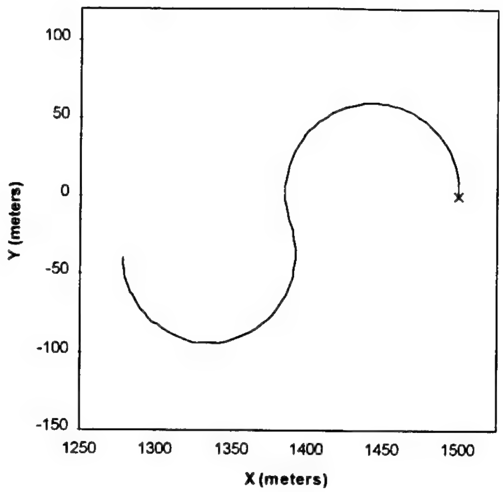
$$\mathbf{F} = e^{\mathbf{A}T} = \begin{bmatrix} 1 & T & (\alpha T - 1 + e^{-\alpha T})/\alpha^2 & 0 & 0 & 0 \\ 0 & 1 & (1+e^{-\alpha T})/\alpha^2 & 0 & 0 & 0 \\ 0 & 0 & e^{-\alpha T} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & (\alpha T - 1 + e^{-\alpha T})/\alpha^2 \\ 0 & 0 & 0 & 0 & 1 & (1+e^{-\alpha T})/\alpha^2 \\ 0 & 0 & 0 & 0 & 0 & e^{-\alpha T} \end{bmatrix}$$

The discrete-time process noise covariance matrix $\mathbf{Q}$ is given by

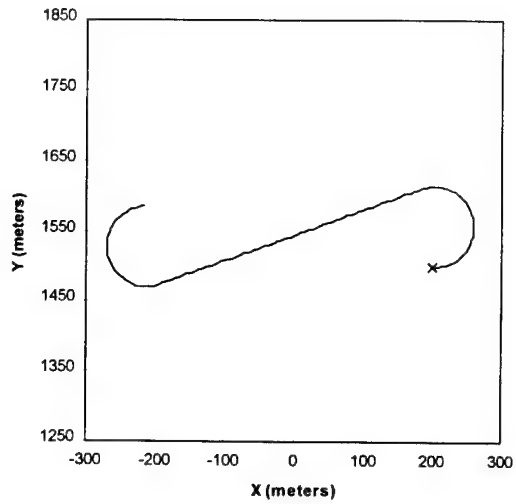
$$\mathbf{Q} = 2\alpha\sigma_m^2 \begin{bmatrix} T^5/20 & T^4/8 & T^3/6 & 0 & 0 & 0 \\ T^4/8 & T^3/6 & T^2/2 & 0 & 0 & 0 \\ T^3/6 & T^2/2 & T & 0 & 0 & 0 \\ 0 & 0 & 0 & T^5/20 & T^4/8 & T^3/6 \\ 0 & 0 & 0 & T^4/8 & T^3/6 & T^2/2 \\ 0 & 0 & 0 & T^3/6 & T^2/2 & T \end{bmatrix} \tag{4-26}$$

4.7 Model Comparisons

The five models described above where tested using Monte-Carlo simulations with 50 replications in order to compare the state estimation performance of each model. Two different target paths [107] were used in the simulations. The first was a target performing an S turn lasting 40 seconds and the second is also a S turn maneuver but with an straight segment between turns and lasts for 80 seconds. The target paths are shown in Figure 4-2 while Table 4-1 provides a summary of the maneuver parameters used in the simulations. Figure 4-2a is the simulated target path for the S turn without the straight segment and Figure 4-2b is the simulated target path for the S turn with the straight segment. The “x” denotes the starting position.



a) S turn without straight segment



b) S turn with straight segment

Figure 4-2: Simulated Target Paths

Table 4-1: Kalman Filter Simulation Parameter Summary

	Scenario	
	S turn without straight segment	S turn with straight segment
Initial *x*, *y* position	(1500 m, 0 m)	(200 m, 1500 m)
Initial polar position	$r = 1500$ m, $\vartheta = 0°$	$r = 1513$ m, $\vartheta = 82.4°$
Initial heading	90°	0°
Duration	40 sec	80 sec
Turn rate	10 m/s for 20 sec −10 m/s for 20 sec	10 m/s for 20 sec 0 m/s for 40 sec −10 m/s for 20 sec
Sample rate	$T = 0.5\ \mathrm{s}^{-1}$	$T = 0.5\ \mathrm{s}^{-1}$
Range measurement variance	$\sigma_r = 10\ \mathrm{m}^2$	$\sigma_r = 10\ \mathrm{m}^2$
Bearing measurement variance	$\sigma_\vartheta = 0.0001\ \mathrm{rad}^2$	$\sigma_\vartheta = 0.0001\ \mathrm{rad}^2$
Maximum acceleration	$A_{\max} = 1.745\ \mathrm{m/s}^2$	$A_{\max} = 1.745\ \mathrm{m/s}^2$
Forward velocity	$v_t = 10\ \mathrm{m/s}$	$v_t = 10\ \mathrm{m/s}$
Maximum turn rate	$r_{\max} = 0.1745\ \mathrm{rad/s}$	$r_{\max} = 0.1745\ \mathrm{rad/s}$
Mean number of changes	$\alpha = 0.05556\ \mathrm{s}^{-1}$	$\alpha = 0.05556\ \mathrm{s}^{-1}$

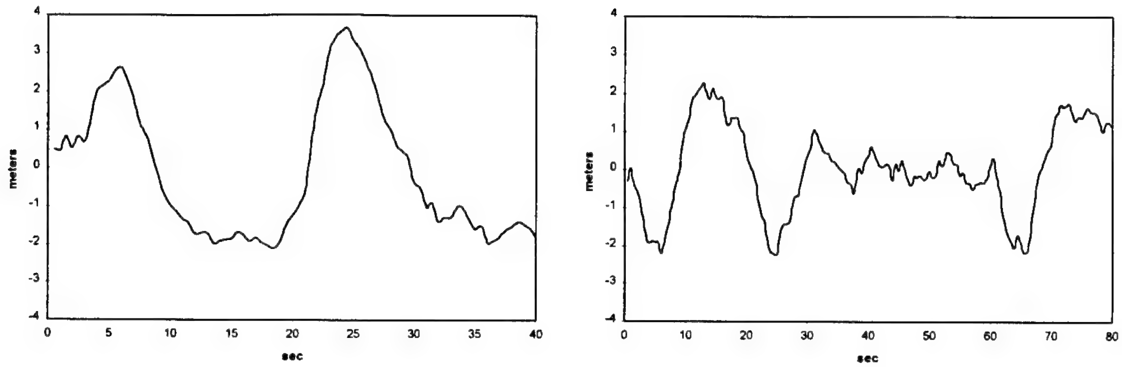
The remaining model specific parameters and initial error covariance matrices needed to perform the filter simulations are as follows:

- Singer (Polar)
  - $P_{max} = 0.1$
  - $P_0 = 0.4$
  - Q as defined in (4-4)
  - P initialized according to (4-7)
- Singer (Cartesian)
  - $\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
  - **P** initialized with [100000 1000 1000 100000 1000 1000] along the main diagonal
- Sklansky
  - $\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
  - **P** initialized with [100000 1000 100000 1000] along the main diagonal
- Helferty
  - $a_1 = 0.1667,\quad a_2 = 0.0249,\quad a_3 = 0.0010,\quad b_1 = 0.2335,\quad b_2 = 0.2132,\quad b_3 = 0.0019$ according to Helferty's formulas [107]
  - $\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
  - **P** initialized with [100000 1000 1000 1000 1000 100000 1000 1000 1000 1000] along the main diagonal
- Bar-Shalom and Fortmann

- $\sigma_m = A_{\max}/6$
- **P** initialized with [100000 1000 1000 100000 1000 1000] along the main diagonal

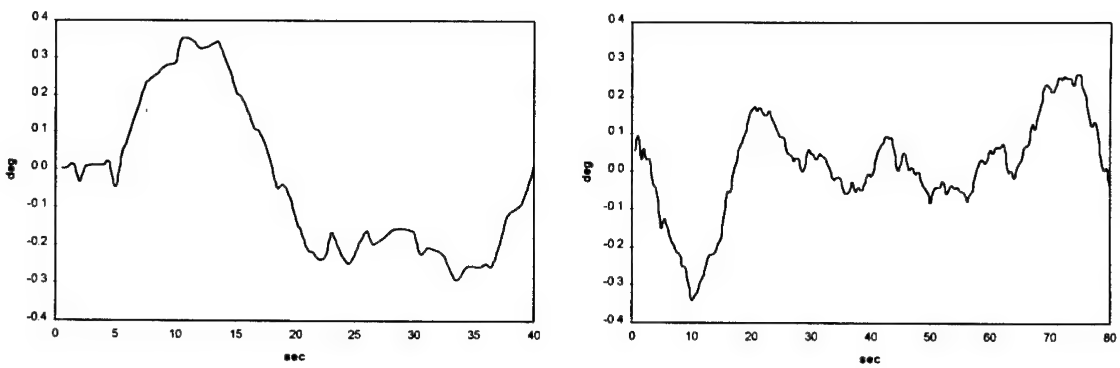
All of the models performed exceedingly well with extremely small average position and velocity errors and RMS position and velocity errors regardless of target path used. Figure 4-3 and Figure 4-4 show the average range and bearing errors, respectively, for both target paths. The average range errors are less than ±4 meters for either target path while the average bearing error is between ±0.3°. The average range and bearing rate errors are show in Figure 4-5 and Figure 4-6 while the RMS range and bearing errors are shown in Figure 4-7 and the RMS range and bearing rate errors are shown in Figure 4-8. The average range rate error is between ±5 m/s and the average bearing rate is between ±0.4 deg/s. The RMS errors are 2-4 meters for range, 0.3-0.6 m/s for range rate, 0.5-2° for bearing and 0.05 deg/s for bearing rate.

The four Cartesian models and the Singer Polar model state estimate converted to Cartesian coordinates are compared next. Since the S turn path is along the 0° radial, the x position error is smaller (±5 m) then the y position (±20 m) for all the models. The opposite is true for the S turn with the straight segment since it is along the 90° radial. The x position error is between ±25 m and the y position error is between ±5 m. This can be seen in Figure 4-9 and Figure 4-10. With few exceptions, the average velocity error, either x or y, are between ±5 m/s. The Singer Polar model with the state estimate converted to Cartesian coordinates and the Sklansky model produce the largest velocity errors but the never exceed ±15 m/s. The average velocity errors are shown in Figure 4-11 and Figure 4-12.



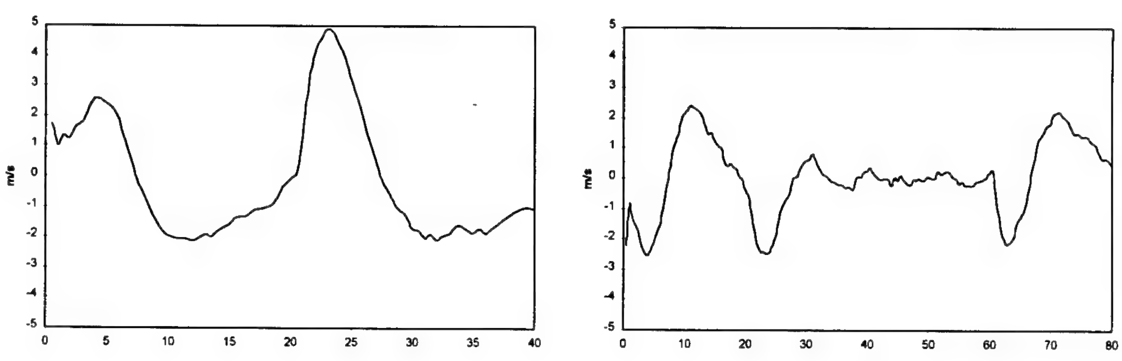
a) S turn without straight segment b) S turn with straight segment

Figure 4-3: Singer Model (Polar) Average Range Error



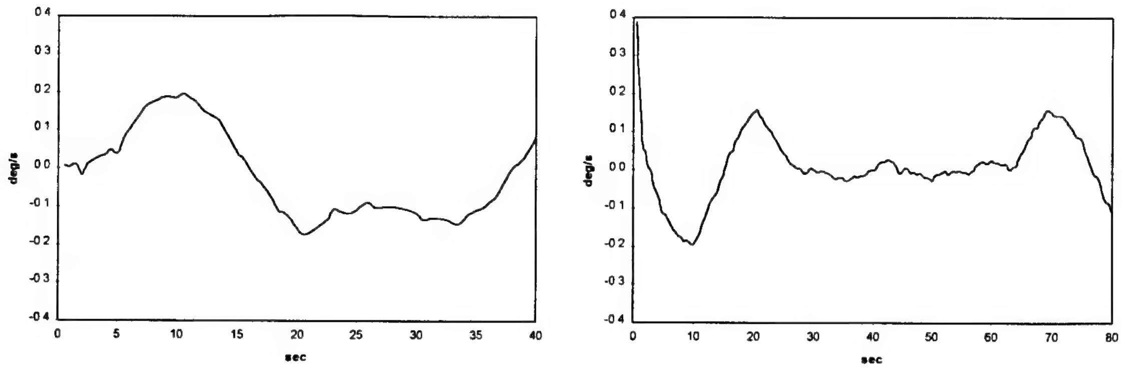
a) S turn without straight segment b) S turn with straight segment

Figure 4-4: Singer Model (Polar) Average Bearing Error



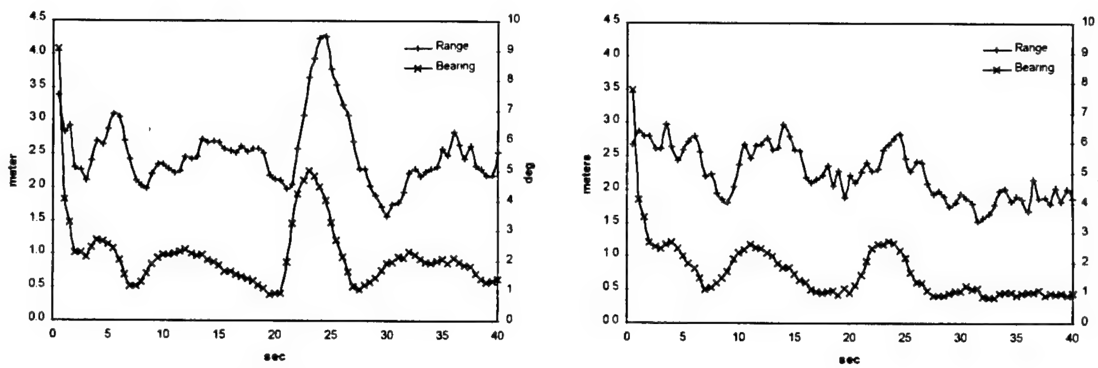
a) S turn without straight segment b) S turn with straight segment

Figure 4-5: Singer Model (Polar) Average Range Rate Error



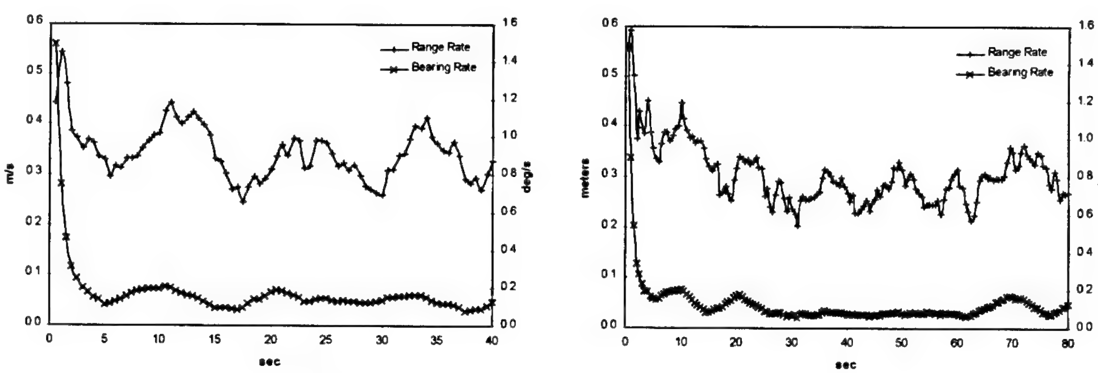
a) S turn without straight segment b) S turn with straight segment

Figure 4-6: Singer Model (Polar) Average Bearing Rate Error



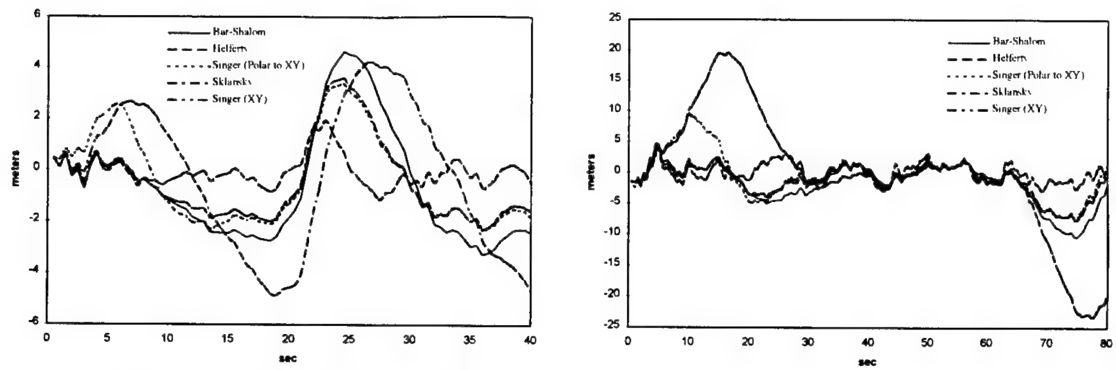
a) S turn without straight segment b) S turn with straight segment

Figure 4-7: Singer Model (Polar) RMS Range and Bearing Error



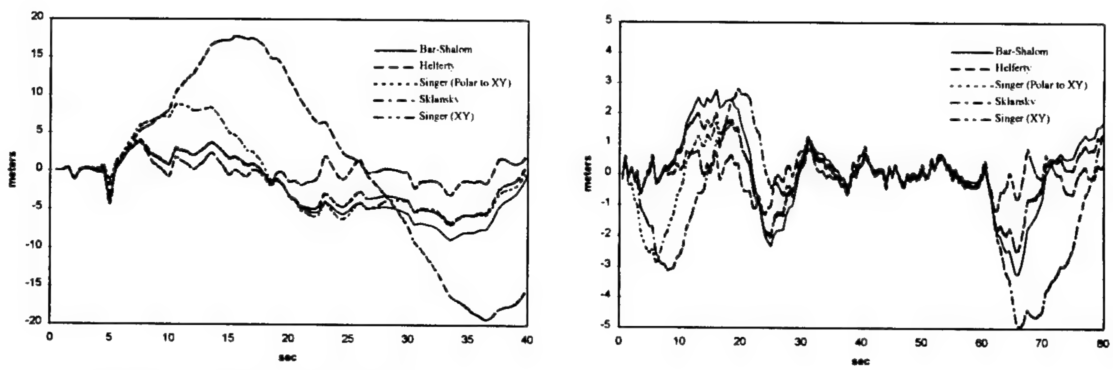
a) S turn without straight segment b) S turn with straight segment

Figure 4-8: Singer Model (Polar) RMS Range Rate and Bearing Rate Error



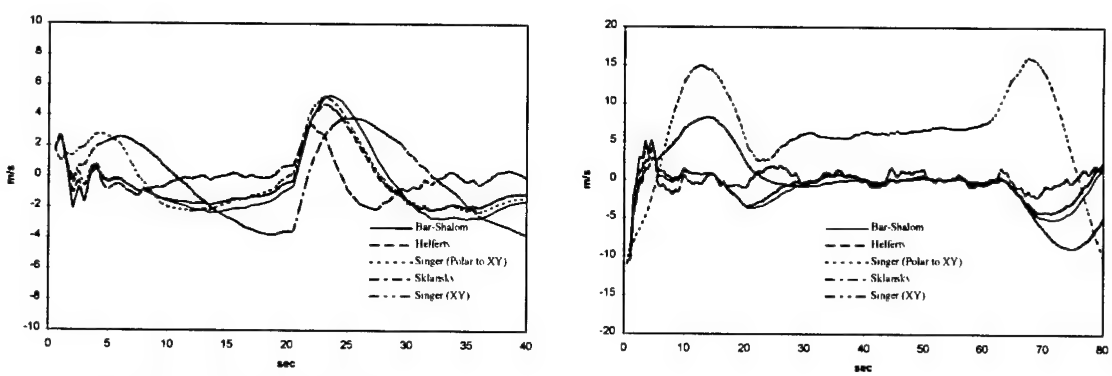
a) S turn without straight segment b) S turn with straight segment

Figure 4-9: Cartesian Models Average X Position Errors



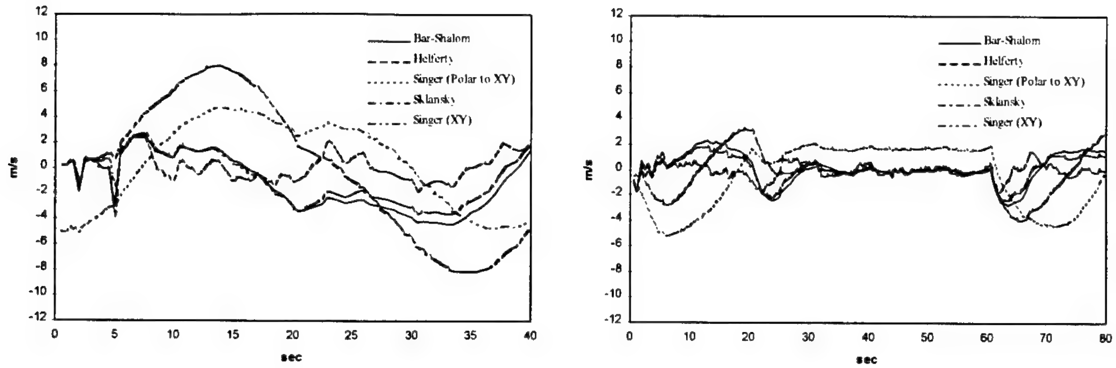
a) S turn without straight segment b) S turn with straight segment

Figure 4-10: Cartesian Models Average Y Position Errors



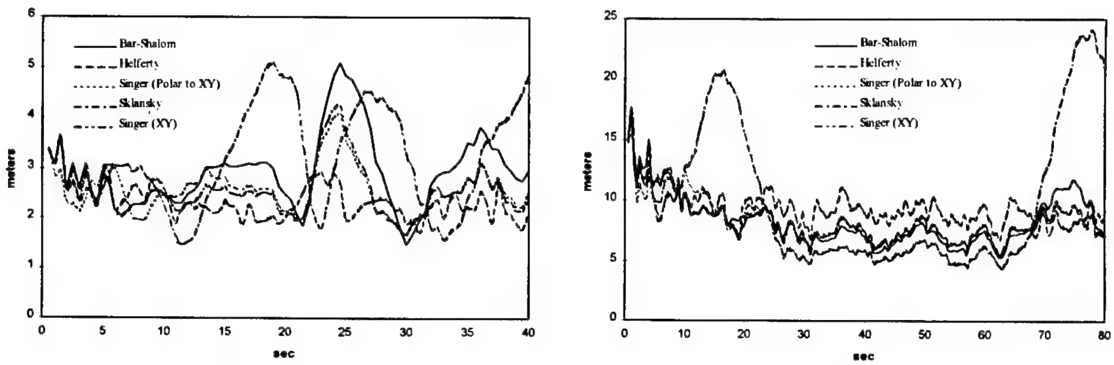
a) S turn without straight segment b) S turn with straight segment

Figure 4-11: Cartesian Models Average X Velocity Errors



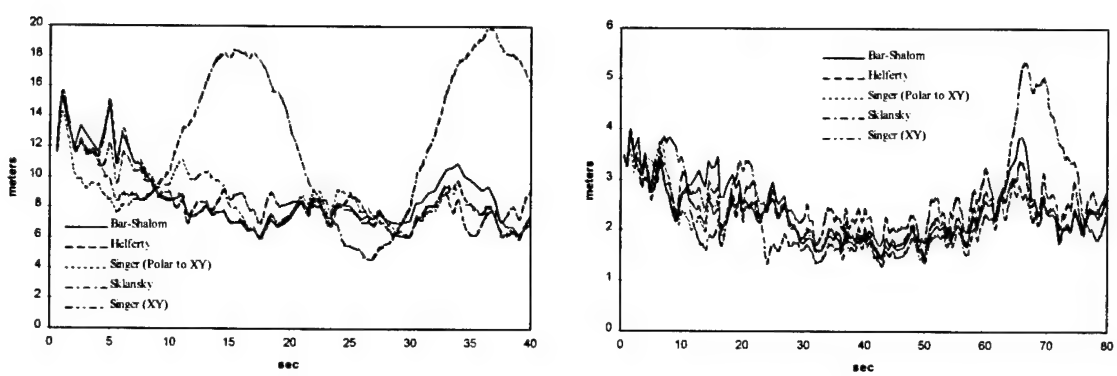
a) S turn without straight segment   b) S turn with straight segment

Figure 4-12: Cartesian Models Average Y Velocity Errors



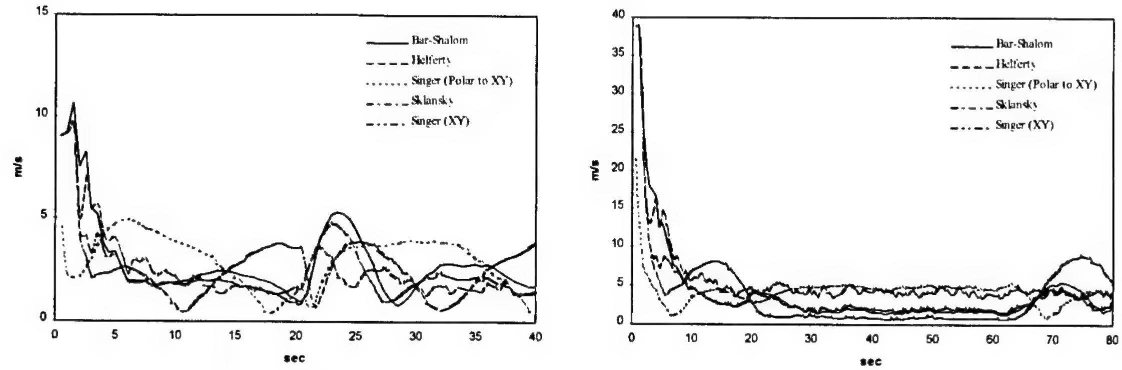
a) S turn without straight segment   b) S turn with straight segment

Figure 4-13: Cartesian Models RMS X Position Errors



a) S turn without straight segment   b) S turn with straight segment

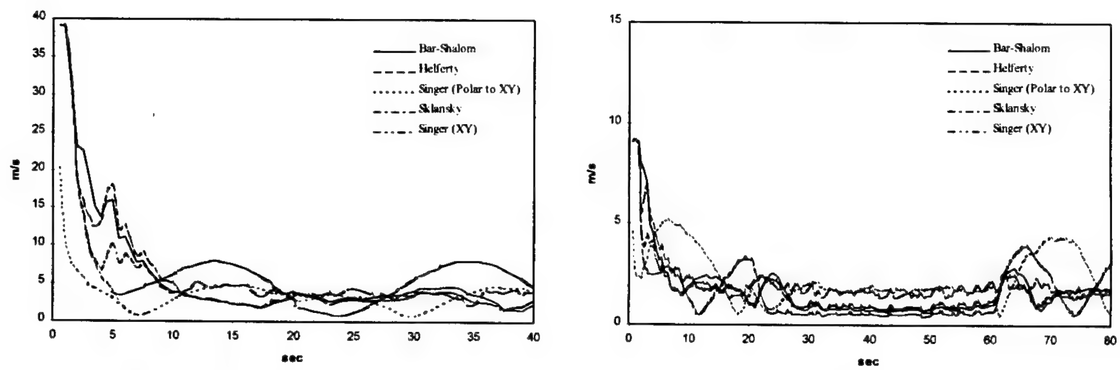
Figure 4-14: Cartesian Models RMS Y Position Errors



a) S turn without straight segment

b) S turn with straight segment

Figure 4-15: Cartesian Models RMS X Velocity Error



a) S turn without straight segment

b) S turn with straight segment

Figure 4-16: Cartesian Models RMS Y Velocity Errors

The RMS errors for both position and velocity are almost indistinguishable. The RMS position errors are shown in Figure 4-13 and Figure 4-14, respectively. The RMS x and y velocity errors are shown in Figure 4-15 and Figure 4-16. As expected, the Sklansky model performs the worst since it is a constant velocity model that does not include acceleration, *e.g.* acceleration treated as added noise.

Table 4-2: Maneuvering Target Model Complexity

Model	
Singer (Polar)	2270
Singer (Cartesian)	2274
Helferty	8390
Sklansky	896
Bar-Shalom and Fortmann	2946

4.8 Summary

The exponentially correlated acceleration models appear to be valid and accurate models of target maneuvers as demonstrated above. All of the model, whether in polar, Cartesian, or polar converted to Cartesian provide very accurate position estimates. The only significant difference is when velocity estimates are considered due to the nonlinear conversion of the Singer Polar estimates to Cartesian estimates and the constant velocity assumption of the Sklanksy model. Besides state estimate accuracy, another consideration in choosing a maneuvering target tracking model is the computational complexity of the model. One such measure is the number of floating point operations (flops).

Table 4-2 shows the number of flops for one iteration of state estimate extrapolation, error covariance extrapolation, Kalman gain matrix computation, state estimate update and error covariance update for each model. The conversion of the measurement noise covariance matrix from polar to Cartesian coordinates only add an additional 32 flops. As can be seen, the two Singer models and the Bar-Shalom and Fortmann models, each a six state estimate model, require approximately the same number of flops. The Bar-Shalom and Fortmann model requires more flops due to the size of the Q and G matrices. The Sklansky model is a four state estimator and requires about 2/3 of the number of flops of the Singer model while the Helferty model is a

10 state estimate model requiring over three times as man flops as the Singer model. The flops were computed for comparable runs of each model averaged over 80 iterations of the update process using MATLAB.

For the purpose of the simulation performed as part of this research, either the Singer model in Cartesian coordinates or polar coordinates with position and velocity converted to Cartesian coordinates will be sufficiently accurate. If increased accuracy is required, several other options are available. The simplest approach is to apply the debiasing methodology by Lerro and Bar-Shalom [108]. They describe a methodology for computing the measurement error covariance matrix in (4-2) differently that they sate insures the true measurement error statistics are used when performing the Polar to Cartesian conversion. Another possible alternative is to use the multiple model approach where multiple models are maintained simultaneously and determine which state estimate to use based upon detecting and estimating the target's maneuvering. Since the performance of the Singer model can degrade during nonmaneuvering portions of a targets trajectory, one could use two different Singer-based model filters with different values of the maneuver variance, σ_m^2, and time correlation, α, and use hypothesis testing to determine when to switch between the two models [109]. When a target is not maneuvering, the Singer model is used to track the target with $\alpha \rightarrow \infty$ and $\sigma_m^2 = 0$. Once a maneuver is detected the a Singer model with a finite α and $\sigma_m^2 \neq 0$ is used. A similar approach is to use filters of different dimensions and switch between them based on maneuver detection. One such approach is the variable dimension filter of Bar-Shalom and Birmiwal [110] in which they use a four state $(x, \dot{x}, y, \dot{y})$ constant velocity model when a target is not maneuvering. Based on a maneuver detection scheme, new state components are added and a constant acceleration model with six

states, ($x,\dot{x},\ddot{x},y,\dot{y},\ddot{y}$), is used. One the maneuver is complete, the four state model is used again.

Two other possible approaches which can be used to increase accuracy are the interacting multiple model (IMM) algorithm and innovations-based approach. The IMM approach consists of a set of several filters which interact through state estimate mixing to track a maneuvering target. Efe and Atherton [111] present on such example of an IMM utilizing adaptive turn rate models while Blair, *et al.* [112], use IMM filtering based on exponentially correlated acceleration models. Blair, *et al.*, use four models in their IMM filter. They include a constant velocity model, a constant acceleration model, an exponentially correlated model with increasing accelerations and an exponentially correlated model with decreasing accelerations.

Chapter 5

Simulation Study

5.1 Model Description

In order to demonstrate and evaluate the proposed Information Theoretic sensor manager, a two-dimensional multiple target, multiple sensor detection, tracking, and identification simulation model has been developed based on the mathematical model shown in Figure 2-3. The sensor manager functions have been partitioned into the Sensor Scheduler and the Information Instantiator as presented in Figure 2-2. The model has been designed to support any reasonable number of targets and sensors. The position observed by each of the sensors can be controlled independently of the other sensors or cooperatively to form a pseudo sensor. Each target is assumed to maneuver independently with target tracking accomplished by using independent Kalman filters based on the Singer model (in Cartesian coordinates) for manned maneuvering targets described in Chapter 4. The simulation architecture is shown in Figure 5-1.

The simulation model was developed with the underlying assumption that surveillance platforms capable of carrying several different types of sensors (radar, IR, ESM, *etc.*) are sent out to surveil the environment. Each sensor's capabilities and performance are modeled through a Kalman filter observation matrix (one for each sensor) and noise variance of their measurements. As discussed earlier, the simulation model captures sensor characteristics in

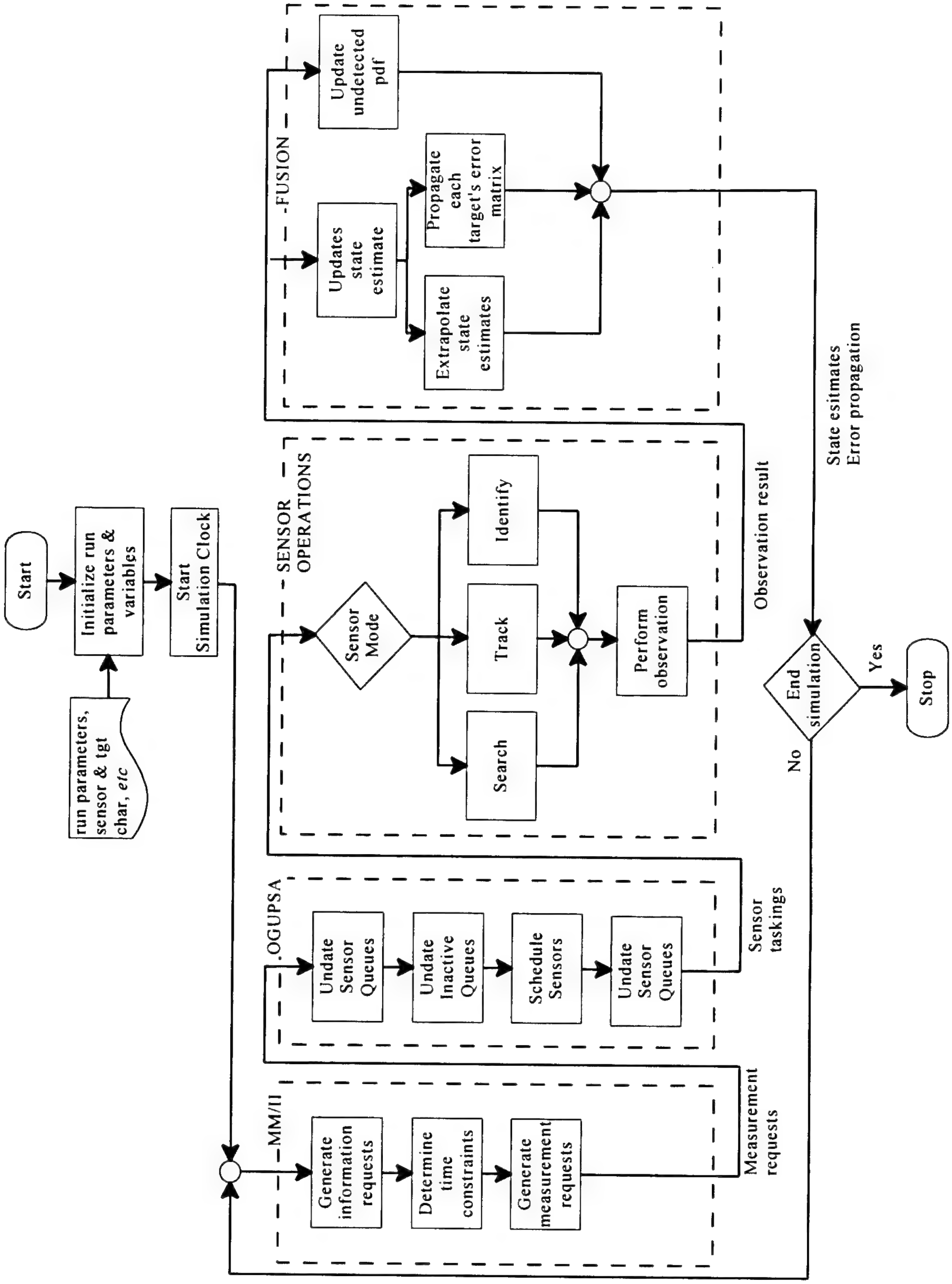


Figure 5-1: Simulation Model Architecture

terms of their P_D, P_{FA}, and fundamental parameter measurement accuracies such as beamwidth, range, range rate, and bearing.

The amount of maneuverability of a target has a direct correlation to the amount of uncertainty about the target's future position. One can either increase the measurement rate of a sensor or combine independent measurements from multiple sensors in order to decrease information or conversely, gain information. In the case of increasing the measurement rate, the amount of information gained is limited by the measurement noise (sensor's accuracy) and the process noise (rate of increase in uncertainty of the target's state due to maneuvers). In the extreme limit, a fixed target yields no new information with each measurement after the first except that gained by averaging repeated noisy measurements. If you are currently tracking a slow maneuvering target that is acting in a predictable manner, it then becomes possible for the sensor manager in general and the Information Instantiator in particular to trade off tracking for search or identification. That is, reducing the frequency of observations of the target while not losing track would not result in any significant reduction in the accuracy of the state estimates.

5.2 Search Area

In order to apply information theoretic measures, the search area is represented probabilistically. That is, a search area is divided into $m * n$ cells with each cell containing a probability of an undetected target being in that cell. Collectively, the cells can be considered as a discrete probability density function (pdf). When a search is performed, the return signal based upon a target location results in an measurement vector and then a detection is determined in the fusion space. After each sensor observation, the pdf is updated thus the pdf is a global estimate

of target location uncertainties and can be used to determine the most likely location of an as yet undetected target and hence where a sensor should search next.

By representing the search area as an undetected target location pdf, the information gained by observing the environment can be computed based on mutual information - the expected change in entropy of the pdf before an observation is taken and after it is taken. This is defined as

$$I = H(\text{before}) - H(\text{after}) \tag{5-1}$$

where H is computed by using the discrete Shannon entropy formula

$$-\sum p(x_i) \log p(x_i) \tag{5-2}$$

Since the pdf is based on sensor observations, the pdf is only an estimate of where targets are (or are not) and not their actual location. Actual locations of detected target are maintained separately for comparison in order to evaluate the effectiveness of the various sensor scheduling schemes.

While this method of dividing the search area into grids provides many benefits, it does come at a significant cost most notably the computational overhead associated with maintaining the undetected pdf. After each sensor observation is completed, the new probability in each grid must be computed and each element of the array representing the undetected target pdf must be updated. The larger the search area the more pdf computations must be done and the more time it takes to run the simulation. For example, if the search area is 100 cells by 100 cells then the number of computation required after a sensor observation is 10^4. A 1000 by 1000 cell pdf would require 10^6 updates after each sensor observation.

5.3 Sensors

Several types of sensors at multiple locations and the use of pseudo-sensors are available in the simulation model. The types of sensors represented include sensors that provide range and bearing (with or without Doppler capability), bearings only sensors, electronic support measure (ESM) sensors, and pseudo-sensors. As stated earlier, a pseudo-sensor is one in which two or more sensors work cooperatively to perform a measurement that neither of them is capable of making by itself. For example, two non-collocated bearings-only sensors can be used to measure the position in 2D space even though each can only observe its line-of-bearing. Thus, pseudo-sensors are used to model the cooperative use of multiple bearings-only sensors located on different platforms to provide range and bearing estimates.

One method of simulating the capabilities of various sensors is to explicitly define such characteristics as

- bandwidth
- wavelength
- duration of waveform
- signal power per pulse
- receiver noise strength
- diameter of radar aperture

However, for the purpose of this simulation a more convenient and simpler method is employed. Regardless of the sensor, the sensor's performance can be captured by its P_D, P_{FA}, and beamwidth. The S/N is determined by the environment that the sensor is operating in (*e.g.* level

of clutter and electronic jamming). Then using this S/N and setting a desired P_D (P_{FA}), the sensor's operating characteristics determines the P_{FA} (P_D) [113]. Thus, these three sensor parameters fully specify the sensor's capabilities.

5.4 Targets

Any number and type of targets can be represented in the model. Target movements are driven by random maneuvers of specified variances based on the Singer target maneuver probability function. Different types of targets can be represented by setting the appropriate maximum acceleration, maneuver correlation coefficient, probability of maximum maneuver (positive and negative acceleration) and probability of no maneuver. Additionally, each target can be initialized with any starting range and bearing. There is no interaction of targets -- that is, each operates independently of each other. The actual locations of each target are maintained for ground truth purposes (*e.g.* to determine if the target is inside the sensor's beam and to determine the probability of detection).

5.5 Target State Estimator

An individual Kalman filter is maintained for each target that is detected. Based on the review and testing of maneuvering target models from the previous chapter, a Singer-based Cartesian coordinate model has been selected for use in the simulation model. The reason for this is to keep the target state estimates in the same coordinate system as the undetected target pdf. A multiple model approach with a bank of three filters using different acceleration and probability of maneuvers has been implemented in the simulation model. If the difference between the measured versus filtered position (the innovations process) reaches a specified

threshold then a different filter can be selected. Identification of a target can also result in selecting a different filter to be used.

As discussed in Chapter 3, the error covariance matrix (**P**), maintained as part of the Kalman filter computation, captures the amount of uncertainty associated with a target's state. This covariance matrix is updated after each observation resulting in a decrease in uncertainty or gain in information. The information gained due to the change in uncertainty about the target's state is calculated using the determinant of the error covariance matrix before ($\mathbf{P}_b$) and after ($\mathbf{P}_a$) the update. Using the continuous version of Shannon's entropy formula and assuming a Normal distribution as discussed in Section 3.6.2, the amount of information gained is based on the norms of $\mathbf{P}_b$ and $\mathbf{P}_a$ as given by (3-5).

5.6 Sensor Scheduler

An enhanced version of the dynamic sensor scheduling algorithm called the On-line, Greedy, Urgency-driven Pre-emptive Scheduling Algorithm (OGUPSA) [19], [20] has been incorporated into the model. OGUPSA was developed using the three main scheduling policies of Most-Urgent-First to pick a task, Earliest-Completed-First to select a sensor, and Least-Versatile-First to resolve ties. One of the key components of OGUPSA is the information in the applicable sensor table. This table is the mechanism that is used to assign requested tasks to specific sensors.

Significant improvements and modifications to OGUPSA have been made in order to implement the algorithm for use in this simulation. Of particular interest is the expansion and

development of the OGUPSA applicable sensor table to more realistic tasks than in the original OGUPSA paper [19]. The original work focused on a unit execution time task scheduling problem without any task preemption. Logic has also been added to insure that a task requiring more than a unit execution time is not interrupted during the performance of a task. Another improvement restricts the scheduling and initiation of a task by using a "commence no sooner than" time. This can be used to schedule future tracking or identification tasks at specific times. The final enhancement involves the use of pseudo-sensors. Two types of pseudo-sensors have been incorporated into OGUPSA. The first is a sensor that operates in several modes. An example of this is a Doppler radar operating using either Doppler or not using it. The other type of pseudo-sensor is the cooperative use of 2 bearings-only sensors at different locations in order to obtain range and bearing measurements of a target. An updated version of the OGUPSA scheduler architecture is shown in Figure 5-2.

5.7 Information Instantiator

The sensor manager is concerned with searching, tracking, and identifying. These manager functions need to be mapped to sensor scheduling tasks. It is the Information Instantiator that determines what observation functions are required based on computed expected information for each request from the mission manager. As discussed in previous chapters, information requests which are passed from the MM to the II are of three types, search, track, and identification. Along with each of these requests is an indicator of the type or amount of information required by the mission manager as well as temporal constraints before which or after which the fulfilling of the request would be of decreased value to the MM. An applicable function table maps the sensor management functions to the tasks used in OGUPSA's applicable sensor table has been

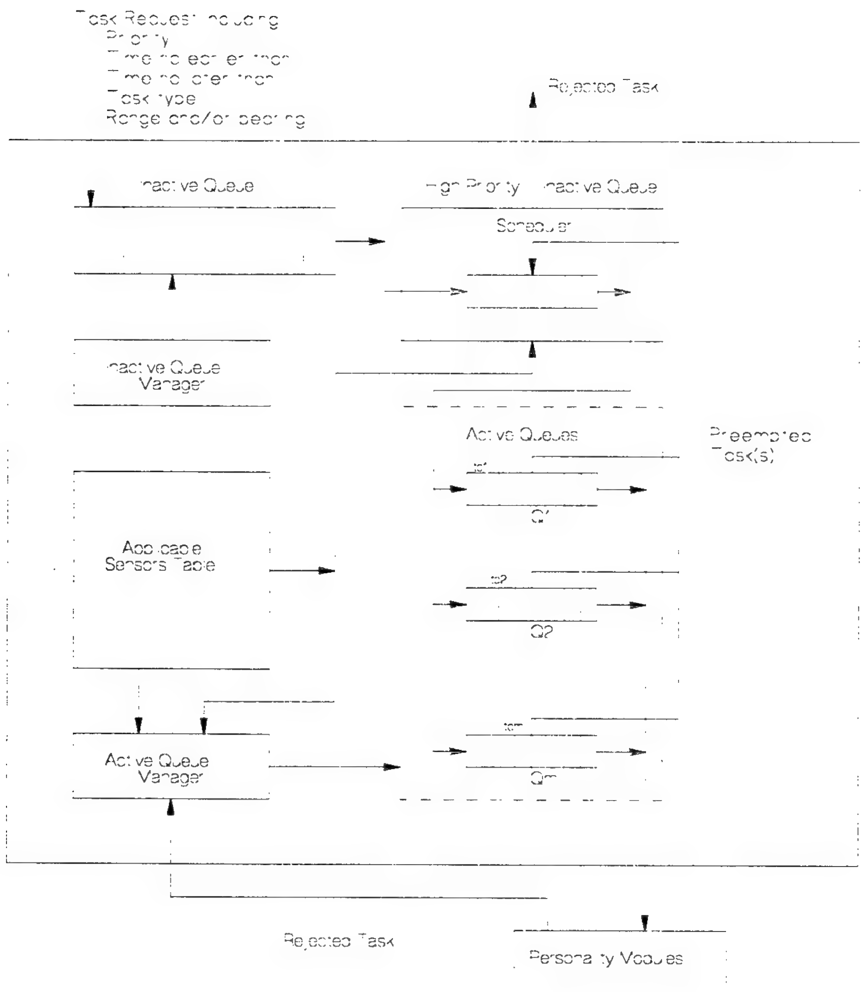


Figure 5-2: Enhanced OGUPSA Scheduler Architecture

developed and implemented. The applicable function table provides the mechanism for the sensor manager to request sensor independent tasks to meet specific mission goals and it becomes the responsibility of the sensor scheduler to assign those tasks to specific sensors. An example of an applicable function table is shown in Table 5-1.

Table 5-1: Applicable Function Table Mapping Management Functions to Scheduling Tasks

Functions	Sensor Scheduling Tasks										
Task	1	2	3	4	5	6	7	8	9	10	11
Accuracy	Low x	High x	Low y	High y	Low x,y	High x,y	Low x,y High $\dot{x}$	High x,y Low $\dot{x}$	Low x,y (x detect)	Low x,y (y detect)	High feature
Search	X		X		X		X				
Transition to Track					X	X			X		X
Track											
High accuracy		X		X		X		X			
Low accuracy	X		X		X		X				
Reacquire							X		X		
Identify											X

5.8 Programming Language

The simulation model was developed on a Sun SPARC workstation and a DEC Alpha workstation. However, the model can also be run on IBM compatible personal computers as well as most computer workstations. For ease of programming, the model was developed using the matrix-based MATLAB programming language. The main drawback of this language is that it is an interpreter so execution can be slow. The major advantage of using MATLAB is its build-in graphics capability and the inherent programming structure that can later be converted to the C-language or a simulation language for compilation and faster execution as well as its portability.

Chapter 6

Simulation Results

6.1 Search Area

The search area used in the following example is assumed to be a 10 km^2 area. The pdf for this search area is divided into 10^6 cells with each cell representing a 10 m^2 area. The center of the search area is assumed to be at a significantly large enough range so that the small angle approximation can be used. That is

$$s = r\,d\theta \tag{6-1}$$

where is s is the arc length, r is the range, and $d\theta$ is the angle in radians. Sensors with beamwidths of 0.006° (100 μrad), 0.1° (1750 μrad), and 1° (17500 μrad) at 100 km would correspond to a beamwidth of 1, 17 and 175 cells respectively assuming linear beamwidths.

6.2 Sensor Description

Five sensors with different P_D, P_{FA} and measurement noise were used to detected, track, and identify targets. Four types of sensor were modeled and include Radar (1 with Doppler capability), forward looking infrared (FLIR), infrared (IR) scanner, and an electronic support measure (ESM) sensor. The sensors are located either along the X or Y axis of the search area so the two locations will be orthogonal to each other. Regardless of which axis the sensor is located on, the sensors are assumed to be 100 km from the center of the search area. The types of

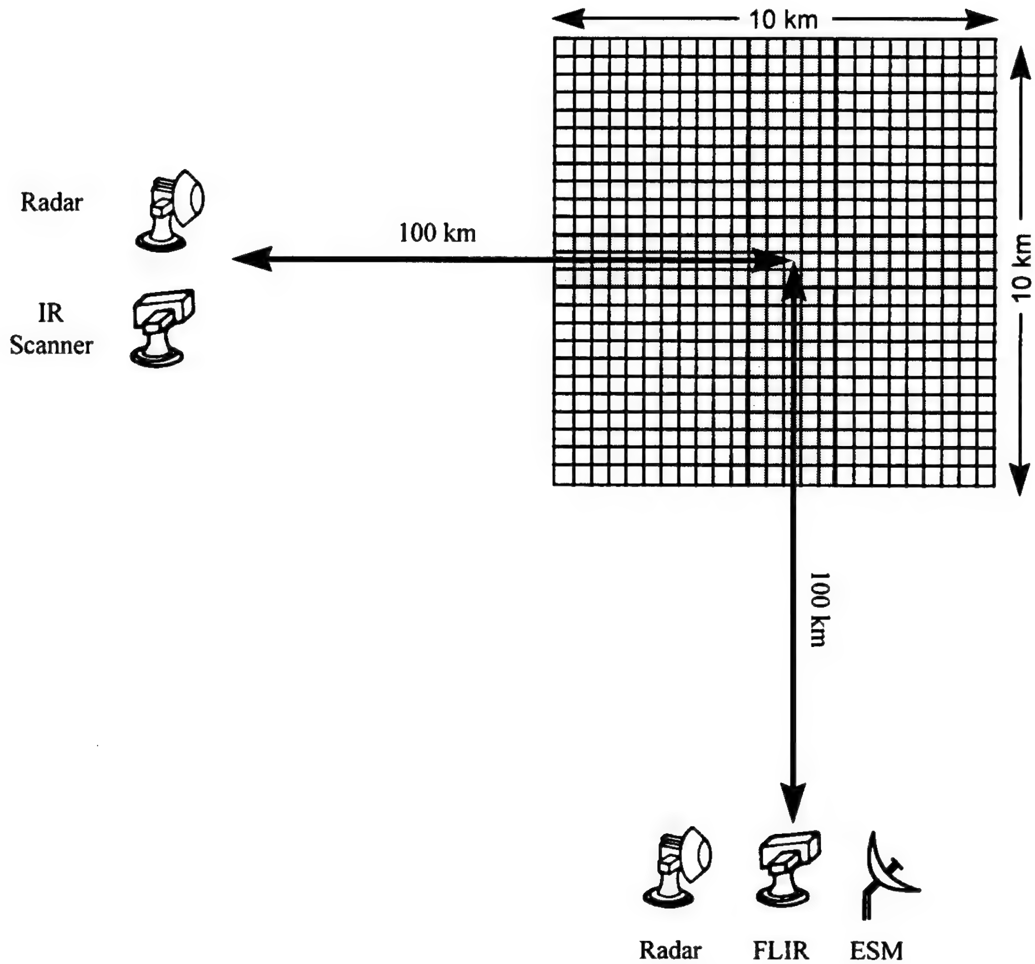


Figure 6-1: Sensor Locations and Search Area Diagram

Table 6-1: Sensor Description

	Sensor A	Sensor B	Sensor C	Sensor D	Sensor E
Nominal type	Doppler Radar	Radar	FLIR	IR scanner	ESM
Characteristics	Range 90m ± 10% (0.9 cell) Bearing 1° = 6σ (29 cells) Range rate ± 10%	Range 30m ± 10% (0.3 cell) Bearing 0.1° = 6σ (2.8 cells)	Bearings-only 0.1° = 6σ (2.8 cells)	Bearings-only 100μrad = 6σ (1 cell)	Bearings-only 1° = 6σ (29 cells)
Location	X axis	Y axis	X axis	Y axis	X axis
P_D	0.95	0.95	0.99	0.99	0.5
P_{FA}	0.001	0.001	0.001	0.001	0.01

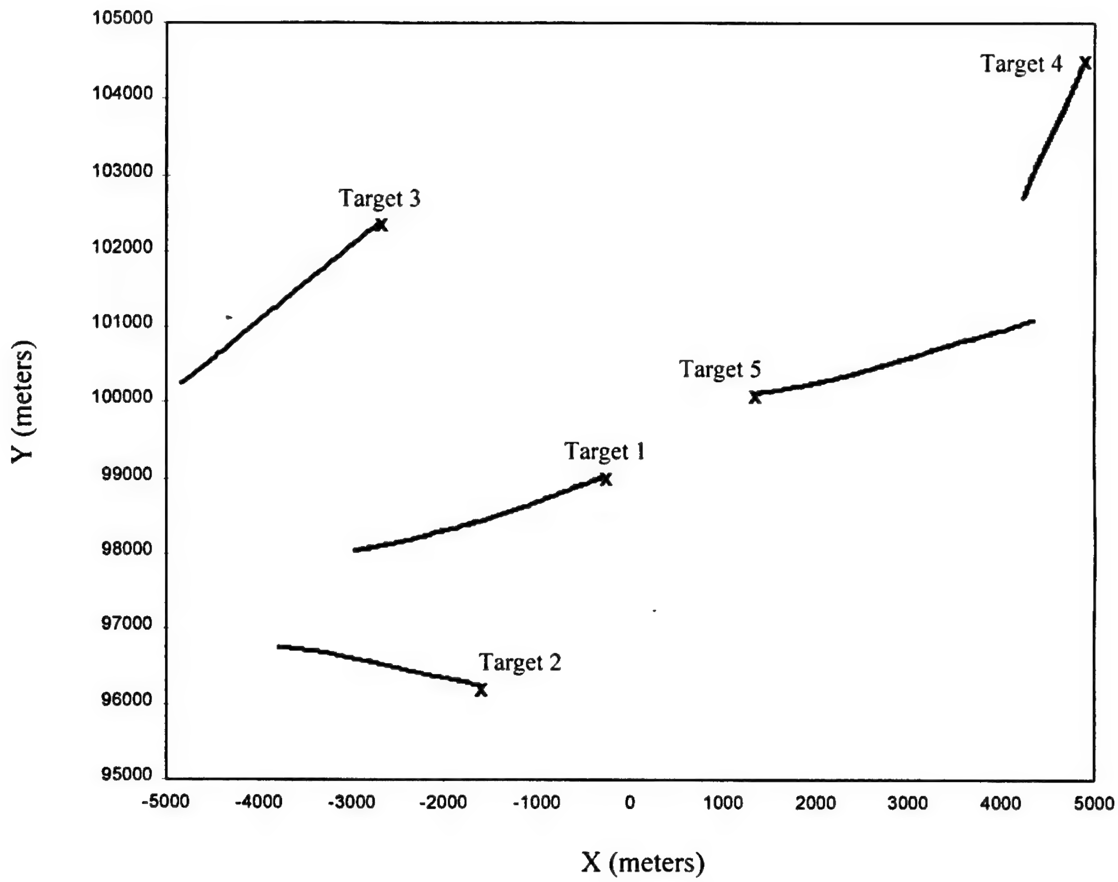


Figure 6-2: Ground Truth of the Targets

sensors and their locations relative to the search area are shown in Figure 6-1 while a descriptive summary of the sensors characteristics is provided in Table 6-1.

6.3 Targets

Three classes of targets classes were modeled and include fighter, bomber, and transport targets. A total of five targets -- 3 fighters, 1 bomber, and 1 transport -- were used. As stated in the previous chapter, each target operates independently of each other so there are no interactions

Table 6-2: "In Harm's Way" Goals

Goal Number	Goal	Included Goals
1	to obtain and maintain air superiority	2, 3, 4, 5
2	to minimize losses	6, 7, 8
3	to minimize personnel losses	6, 7, 8
4	to minimize weapons expenditure	6, 8
5	to seize the element of surprise	8
6	to avoid own detection	9, 10
7	to minimize fuel usage	10, 11
8	to minimize the uncertainty about the environment	12, 13
9	to navigate	15, 16
10	to avoid threats	15, 16
11	to route plan	15, 17
12	to maintain currency of the enemy order of battle	14, 16
13	to assess state of the enemy's readiness	14
14	to collect intelligence	15, 16, 17
15	to track all detected targets	
16	to identify targets	
17	to search for enemy targets	

between targets. Figure 6-2 shows the paths of each target's ground truth with the position of the targets at the beginning of the simulation runs denoted with an "x."

6.4 Lattice of Goals for Determining Weights Used by the Mission Manager

Since an "in harm's way" scenario assumption is being used, a subset of applicable Air Force goals from Figure 2-10 were identified and used to produce a lattice of goals. Seventeen of the 90 Air Force goals that apply to the "in harm's way" assumption were used to produce lattice that can be described as a simpler, pruned version of the entire Air Force lattice. The goals that were used are listed in Table 6-2 with the resulting lattice and associated weights for each goal shown in Figure 6-3. The bottom three goals (observation functions) are track, identify, and search with weights of 0.36, 0.46, and 0.18 respectively. These weights were then

used to establish the priority associated with the observation tasks sent to OGUPSA (the sensor scheduler).

6.5 Sensor Management Comparisons

As discussed in Chapter 2, previous sensor managers have been based on *ad hoc* methodologies. This is the first mathematically rigorous sensor management model and as such, there are no other sensor management schemes to compare it with. In an attempt to perform a comparison, the simulation was run using a purely random sensor management scheme and the sensor management methodology (including the Mission Manager and Information Instantiator) presented in this dissertation. In both cases the OGUPSA sensor scheduling algorithm was used to schedule tasks to sensors. In the random case, the weights for the three functions (search, track, id) were all equal and the search aimpoints were chosen randomly along with the time between track updates.

For the information theory based sensor manager, the weights from the lattice in Figure 6-3 was used to set the priorities for the three functions. For search tasks, the pdf cell with the highest probability of an undetected target was chosen as the aimpoint for the sensor. In the case of tracking tasks, an information threshold was defined and the target error covariance matrix was extrapolated to estimate the information rate in order to determine when to perform a track update. Lastly, an identification tasks was requested once a target track had been established.

The simulation was run for 100 time intervals with each time increment equal to 0.1 seconds. As expected the random case did not perform well. Without using weights and randomly

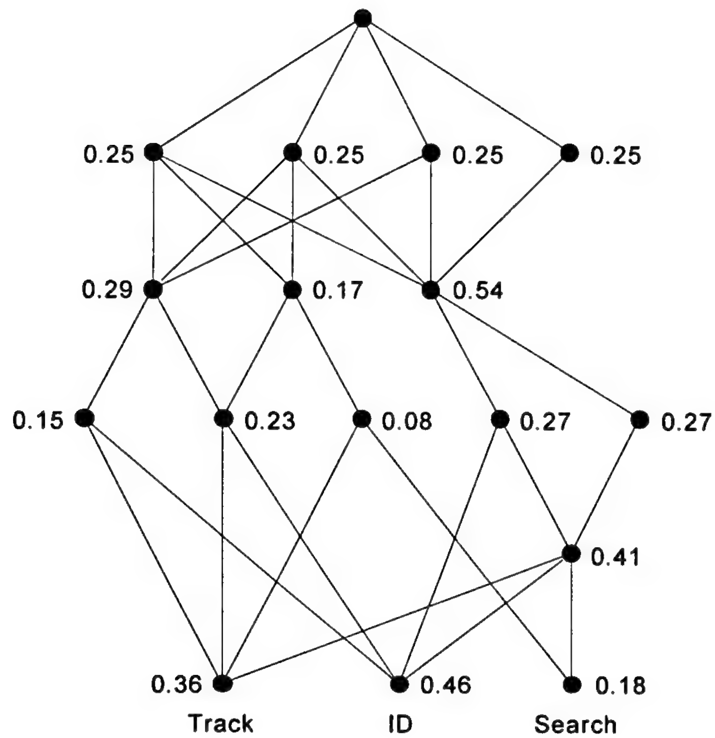


Figure 6-3: "In Harm's Way" Lattice

choosing where to search, targets took longer to detect and establish track due to the rejection of transition to track and track requests by OGUPSA. This demonstrates the need for a method, such as the use of POSETs and lattices, to establish weights that can be used to establish priorities between the search, track, and identify functions.

Generally, the information based sensor manager detected and established track on all 5 targets sooner that the random approach. But looking at the change in entropy of the search area as shown in Figure 6-4, there is no significant difference between the two cases. The reason for this is that the number of sensor operations for each sensor in both runs were approximately equal -- the information theory based simulation was just more efficient in detecting and establishing tracks of detected targets than the random simulation.

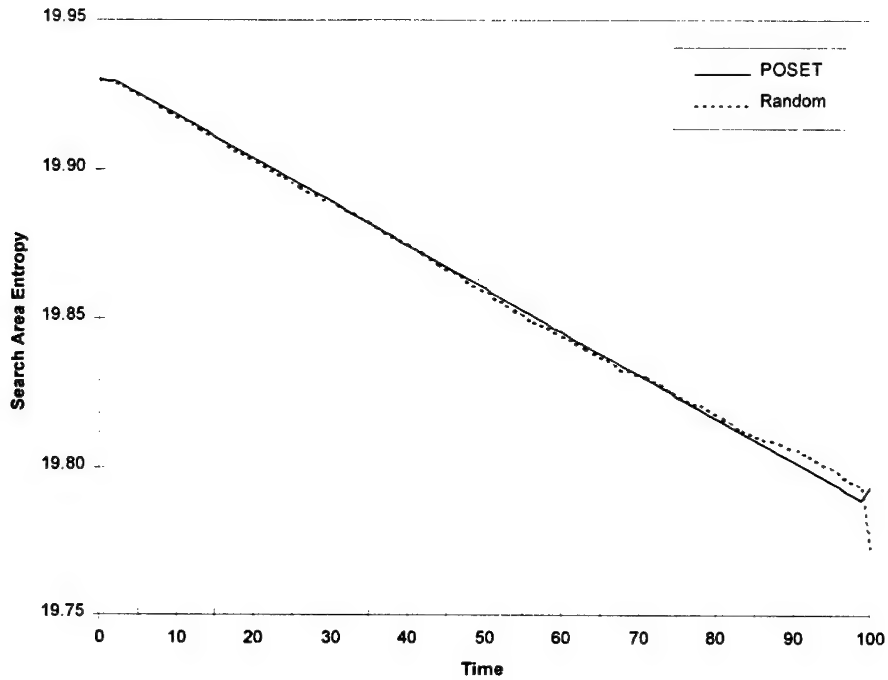
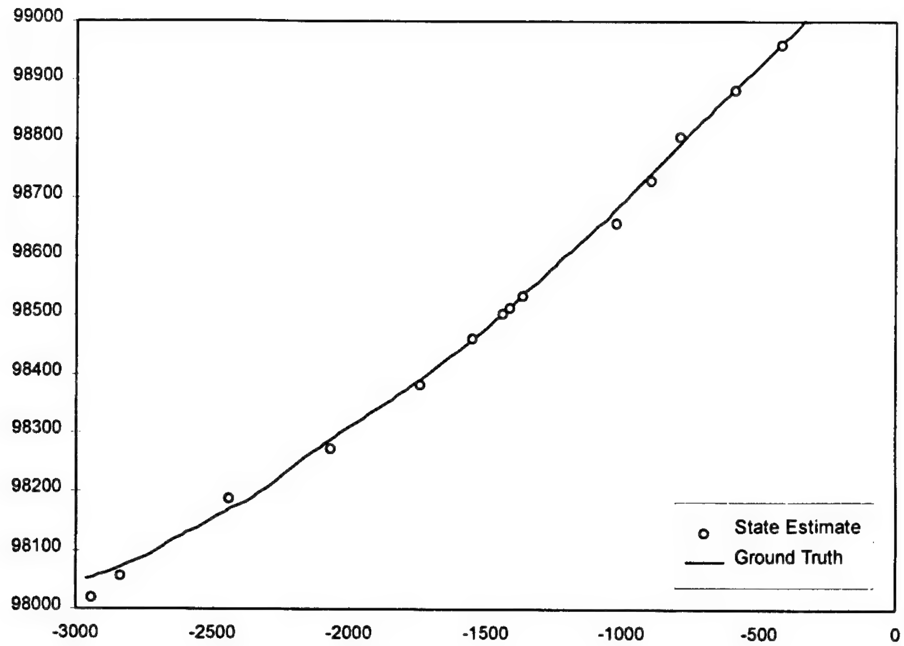


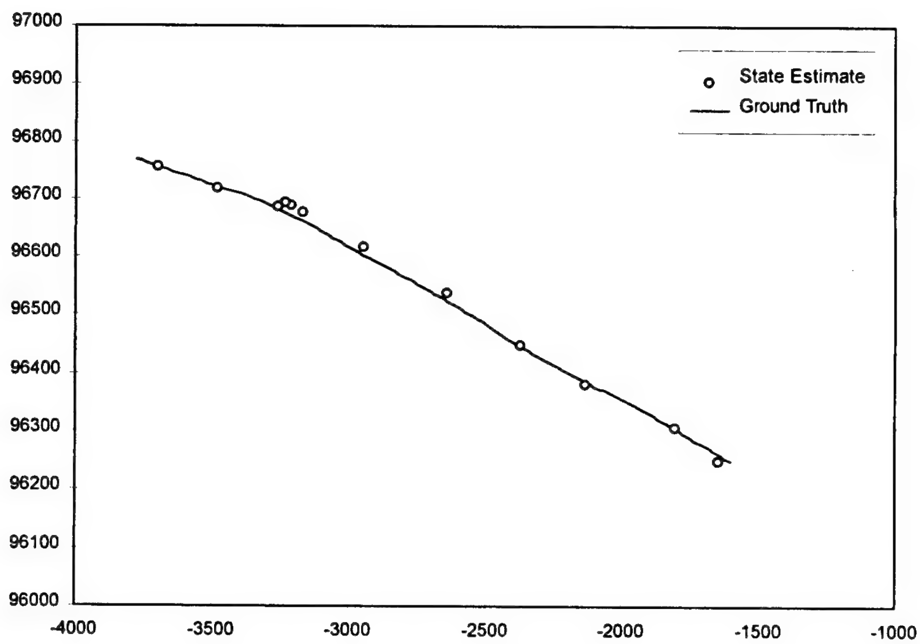
Figure 6-4: Change in Search Area Entropy

Once tracks were established for a target, the state estimation process performed equally well for both management schemes. As stated previously, the Cartesian coordinate version of the Singer model was used to provide the kinematic state estimates for the targets. The updated state estimate, $\hat{\mathbf{x}}_k^+$, for targets 1 (fighter) and 2 (bomber) are presented in Figure 6-5. As can be seen, the Singer-based model Kalman filter performed extremely well.

As discussed earlier, the update rate of a target in track is dependent on the change in uncertainty, captured by entropy, reaching a specified threshold. Once a track has been established for a target, an initial error covariance matrix, **P**, is established. **P** continues to grow until an update of the target's state estimate is made via a sensor observation. The observation is

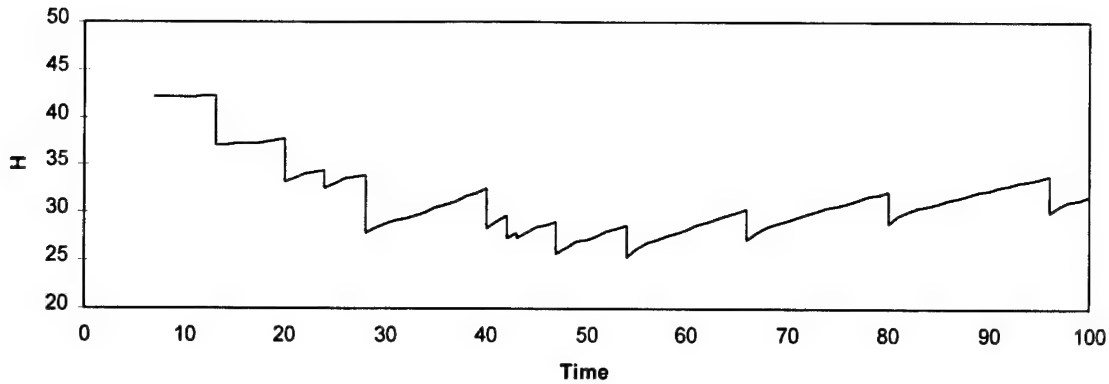


a) Target 1 (Fighter)

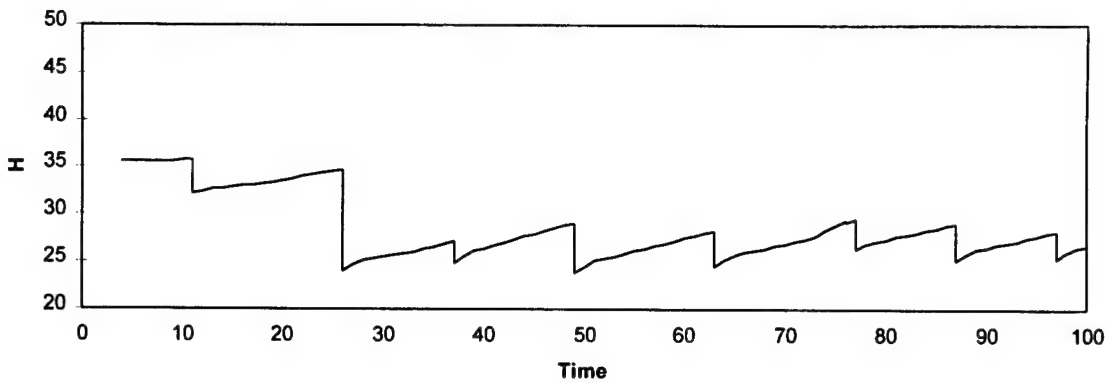


b) Target 2 (Bomber)

Figure 6-5: State Estimate Performance



a) Target 1



b) Target 2

Figure 6-6: Change in Target State Estimate Uncertainty as Measured by the Entropy, H, of the Error Covariance Matrix, P

converted to a measurement and the target's state estimate is updated. This update process reduces the error associated with the target's state estimate -- the **P** matrix is reduced. This can easily be seen in Figure 6-6 where entropy of the **P** matrix is plotted for targets 1 and 2. The peaks are the extrapolated **P** matrix prior to a sensor measurement, $\mathbf{P}_k^-$, and the point directly below is the updated **P** matrix, $\mathbf{P}_k^+$, after the sensor measurement. The difference between $\mathbf{P}_k^-$ and $\mathbf{P}_k^+$ is the amount of information gained by the sensor measurement.

As previously discussed in this section, it is virtually impossible to compare different sensor management schemes due to their *ad hoc* nature. As such, two sensor management schemes - Goal Lattice (information based sensor management system) and Non-Prioritized (with random search and random time between track) - were run in order to demonstrate the use of goal lattices. A summary highlighting the differences between the proposed information based sensor management and the non-prioritized, random sensor manager is shown in Table 6-3, Table 6-4, Table 6-5, and Table 6-6. Regardless of the measure of effectiveness, the new Goal Lattice system performed superior to the non-prioritized one. The Goal Lattice system initialized track on average half as fast as the non-prioritized system and transitioned detection of targets to track nearly an order of magnitude sooner (Table 6-3). While the non-prioritized system always had failures of when transitioning a detection to track (Table 6-4) and occasionally had reacquiring track failures after a track update was missed (Table 6-5), the goal lattice system never did. Additionally, as shown in Table 6-6, the goal lattice system always had all of the targets in track at the end of the simulation.

6.6 Summary

The simulation model has demonstrated the use a new sensor management methodology that utilizes POSETs to weight mission goals used by the mission manager to prioritize sensor tasking coupled with an information theoretic based sensor manager. POSETs provide a mathematically traceable methodology to establishing priorities that can be used by the sensor scheduler (OGUPSA) to schedule a suite of sensors to meet the goals of a mission. Additionally, the use of Information Theory provides a mathematical foundation used by the Information

Table 6-3: Summary of the Track initialization Results of Non Prioritized and Goal Lattice Sensor Management

	Average Track Initialization		Average Interval Between Detection and Track Initialization	
Target	Non Prioritized	Goal Lattice	Non Prioritized	Goal Lattice
1	28.0	6.8	13.6	2.0
2	23.9	18.1	17.1	2.7
3	32.9	32.5	17.5	4.3
4	46.8	18.4	32.9	2.7
5	25.3	11.9	19.1	4.0
Average	31.4	17.5	20.3	3.1

Table 6-4: Transition to Track Failure Results of Non Prioritized and Goal Lattice Sensor Management

Transition to Track Failures	Non Prioritized	Goal Lattice
Minimum	2	0
Maximum	19	0
Average	8.2	0

Table 6-5: Reacquire Track Failure Results of Non Prioritized and Goal Lattice Sensor Management

Reacquire Track Failures	Non Prioritized	Goal Lattice
Minimum	0	0
Maximum	2	0
Average	0.6	0

Table 6-6: Targets in Track at End of Simulation Result of Non Prioritized and Goal Lattice Sensor Management

Targets in Track at End of Simulation	Non Prioritized	Goal Lattice
Minimum	2	5
Maximum	5	5
Average	4.3	5.0

Instantiator to determine when to request an update to a target's state estimate. Difficulty in completely evaluating this new approach arises from sensor management approaches that are not well defined. The lack of mathematically based sensor management architectures prevent comparisons and evaluation of the performance of the methodology described in this dissertation. However, a framework now exists for evaluating alternative methods for sensor scheduling, information instantiation, mission management, and sensor fusion.

Chapter 7

Summary and Conclusions

While several sensor management approaches have been proposed in the literature, all appear to suffer from the mixing of sensor physical requirements with information needs. What has resulted is a comingling of not only noncommensurate but inappropriate measures leading to *ad hoc* methods of sensor management. The dissertation presents a new, original hierarchical sensor management model predicated on information theoretic measures and partially ordered sets (POSET). Using the expected change in entropy, expected information gain has been shown to be a valid approach to sensor management in order to trade-off such functions as search, track, and identify.

While using information gain is a necessary condition, it is not a sufficient condition for complete sensor management. Information gain can be used to perform sensor management trade-offs but it does not take into account the multiplicity of competing mission goals. The approach developed here and demonstrated through a simulation model which overcomes this limitation, is based on the use of inclusion relationships among the goals and partially ordered sets of these goals. This facilitated the construction of a hierarchy of goals using a mathematical means to weight the multiple, competing goals thus establishing a means to prioritize the sensor management functions and sensor actions. This methodology can be applied to both military and

civilian situations resulting in a new, quantitative, and traceable measure of importance that a sensor manager can use to perform and optimize trade-off among the sensor management functions.

Chapter 1 described the motivation for this research along with the problem definition. Also, applications of sensor management were described including the "in harm's way" scenario, the search and rescue endeavor of NASA, the management of several low earth orbit satellites to maintain space object ephemeris, and data mining of large databases.

In order to better understand and define the role of sensor management, a comprehensive review of current literature was presented in Chapter 2. Basically, sensor management is a process that performed properly can improve the data fusion process and ultimately our perceptions through the management and coordination of sensor resources. As a result of this literature review, a new comprehensive, mathematically rigorous sensor system model was developed to capture the sensor management process.

Based on this model, an original sensor management system was developed where a Mission Manager (MM) and a Sensor Manager interact within the Information Space. The MM relies on the weights developed from the lattice of mission goals and inputs from human operators to compute information requests and passes them to the Sensor Manager. The Sensor Manager maps the information requests to observation requests and then ultimately schedules tasks to specific sensors. The Sensor Manager subsumes two separate, distinct, and essentially orthogonal tasks allowing the sensor manager to be partitioned into the Information Instantiator

(II) and the Sensor Scheduler. The II converts the information requests from the mission manager into observation requests and passes the observation requests to the Sensor Scheduler where sensor measurements are optimally scheduled.

In developing and simulating the sensor management model, techniques from several disciplines where used. An extension of POSETs and lattices from abstract algebra, called goal-lattices, provides the methodology to order and weight the mission goals and were described in Chapter 2. Chapter 3 provided a background on the uses of information theory as applied to Kalman filtering, data fusion, and sensor management and scheduling. At the conclusion of the chapter the proposed information measures were developed. The use of Kalman filtering and the comparison of several exponentially correlated acceleration models were presented in Chapter 4.

Finally, a simulation model was developed to demonstrate this new sensor management model and described in detail in Chapter 5. The results of the simulation model were then presented in Chapter 6.

7.1 Contributions

Previous approaches to sensor management have treated the problem as a single optimization task with a performance measure as a weighted sum of diverse, noncommensurate measures. The approach developed in this dissertation uses POSETs with superimposed value apportionment in order to provide a quantitative and traceable measure of importance (weights) that a sensor manager can use to perform and optimize trade-off among competing management

functions -- *e.g.* search, track, and identify. Another advantage is that these weights can vary as a function of time or phase of a mission. Different goals are preferred over others and these change during different phases of a mission in response to changes in the environment. A linear transformation approach was used to map the m-dimensional vector of top level goals to a n-dimensional vector of goal values for the competing management functions. Properties of the goal lattice were also presented including value and structural sensitivity. This new sensor management system provides a mathematically based methodology to change the preferences in real-time during a mission based on changes in information produced by data fusion, a human operator, or both.

Past sensor management approaches have been *ad hoc* which makes it difficult to compare different sensor management schemes. This dissertations has developed a hierarchical, mathematical sensor manager and demonstrated its use in a simulation. The results from the simulation suggest that this new model is valid but it was only tested against a random sensor management scheme. However, it did successfully demonstrate the hierarchical approach to sensor management using a mission manager based on weighting of goals coupled with partitioning the sensor management problem into orthogonal tasks (the information instantiator and the sensor scheduler). The simulation based on this new model also highlights the interaction between the sensors, data fusion, mission management, and sensor management. This new sensor management model along with the simulation model provides a basis to compare future management approaches.

7.2 Future Research

There are several interesting directions one might pursue in extending both the sensor management model and the simulation model itself. The first is the way that the maximum time between track updates is computed. The maximum level of uncertainty (a threshold) which is not to be exceed was used in the simulation. While this provides an estimate of the interval between track updates, the closed form method described in Chapter 3 should be investigated. Another follow-on to this research would be the development of a closed loop transfer function of the sensor management system that would allow one to investigate global stability. Further investigation of the goal lattice sensitivity needs to be done. One possibility is to develop a method to identify classes of goal lattices by converting them to a "behaviorally equivalent" lattice using techniques from Sequential Machine Theory.

Additional work needs to be done on the simulation model also. Different data fusion methodologies from the literature need to be reviewed for possible inclusion in the model. This would allow different data fusion approaches (*e.g.* Bayesian versus Dempster-Shafer or centralized versus decentralized) to be studied in concert with different sensor management models.

Another model improvement would remove the limitation on sensor locations. The model could be enhanced to handle sensors at any location and not limit them to being located on orthogonal axes. Lastly, a better method of representing the undetected target pdf would significantly improve the simulation run time. Continually updating 10^6 is computationally expensive. An analog representation, *e.g.* a phosphor screen, of the search area (undetected

target pdf) could be used in a real-time system. As a sensor observes a portion of the search area, the intensity of the corresponding location on the screen increases while areas not searched would decrease in intensity. This screen intensity could then be processed to determine future search locations and entropy calculations to measure the increase of information due to sensor observations.

Appendices

Appendix 1

Below are a list of goals used in developing the NASA POSET and lattice. The goals are based on a combination of NASA goals documented in their strategic plan and goals added by the author. The first column is the node number assigned to the goal (numbered left to right and top to bottom) stated in column 2. The third column is a list of goals included in the goal.

Goal Number	Goal	Included Goals
1	to explore, use, and enable the development of space for human enterprise	4, 5
2	to use the environment of space for research	6, 7, 8, 9
3	to enable technology development and transfer	10, 11
4	to conduct human and robotic missions to planets and other bodies in our solar system to enable human expansion	12, 13
5	to provide safe and affordable human access to space	14, 15 ,16
6	to share knowledge of the Earth system and mysteries of the universe	17, 18
7	to create an international capability to forecast and assess the health of the Earth system	19
8	to create a virtual presence throughout our solar system	20
9	to support research endeavors in space and on Earth	20, 21
10	To develop cutting-edge aeronautics and space systems technologies	22, 23
11	To support the maturation of aerospace industries	24, 25, 26
12	to conduct human missions of exploration of other bodies in the solar system	43
13	to enable future exploration beyond Earth's orbit	43
14	to enable the full commercial potential of space	27
15	to establish a human presence in space	43
16	to share the human experience of being in space	
17	to aid in achieving the science, math and technology goals of the U.S.	43
18	to disseminate information about the Earth system	
19	to advance the scientific knowledge and understanding of the Earth, solar system, and the universe	28

Goal Number	Goal	Included Goals
20	to use the environment of space to expand scientific knowledge	29, 30, 31, 32
21	to expand science knowledge through the use of human capabilities in the space environment	29, 30, 31, 32
22	to enable U.S. leadership in global civil aviation through safer, cleaner, quieter, and more affordable air travel	33
23	to revolutionize air travel and the way in which aircraft are designed, built and operated	34, 35, 36
24	to enable or provide aerospace R&D services, facilities and expertise	37
25	to promote the commercial development of space	
26	to enable the productive use of science and technology in the public and private sectors	
27	to reduce the cost of access to space	43
28	to preserve the environment by studying the Earth as a planet and as a system	38, 39
29	to search for life beyond Earth	43
30	to explore the universe to enrich human life	43
31	to discover planets around other stars	
32	to solve mysteries of the universe	40
33	to preserve our freedoms for future generations	
34	to share knowledge and technologies to enhance the quality of life on Earth	41
35	to conduct aeronautic and space research	
36	to apply new aeronautic and space system technologies	42
37	to enable the expansion of space research and explorations	
38	to increase our understanding of the effect of natural and human-induced activities on Earth	
39	to develop predictive environmental, climate, and natural disaster models	
40	to chart the evolution of the universe and understand its galaxies, stars, planets and life	43
41	to transfer innovative space technologies	43
42	to test space technology	43
43	to increase knowledge of Mars	44, 45
44	to determine if humans can live on Mars	46
45	to determine if life on Mars exists	47, 48
46	to find suitable site for settlement	47
47	to explore Mars	49, 50
48	to analyze samples of mars	50
49	to measure as much of surface as possible	51, 52
50	to navigate	53, 54
51	to maximize duration of mission	55
52	to assess mineral content	56
53	to plan path	57

Goal Number	**Goal**	**Included Goals**
54	to avoid obstacles	57, 58
55	to conserve on-board resources	59, 60, 61, 62
56	to verify data taken by other means	59, 60, 61
57	to avoid stationary obstacles	61, 62
58	to avoid moving obstacles	62
59	to analyze the atmosphere of Mars	
60	to analyze sample	
61	to search for objects	
62	to track objects	

Appendix 2

Below are a list of goals used in developing the USAF POSET and lattice. The goals are based on several USAF and Joint Chief of Staff doctrine manuals and course material from the USAF's Air Command and Staff College material. The first column is the node number assigned to the goal (numbered left to right and top to bottom) stated in column 2. The third column is a list of goals included in the goal.

Goal Number	Goal	Included Goals
0	to compel adversary to due our will	1
1	to achieve control of the air	2, 3, 4, 5
2	to deny enemy freedom to carry out offensive operations	6, 7, 8
3	to obtain and maintain air superiority	9, 10, 11, 12
4	to allow friendly forces to perform their mission	13
5	to control tempo of battle operations	14, 15,16
6	to defend lines of communication	17, 18
7	to protect bases	17, 18
8	to protect forces	17, 18
9	to minimize losses	19, 29, 21
10	to minimize personnel losses	21
11	to minimize weapons expenditure	21
12	to seize the initiative with concentration of forces	22, 23, 24, 25
13	to protect friendly aircraft enroute to their target(s)	39, 40
14	to neutralize units not yet engaged by land forces	26, 27, 28
15	to support surface forces in the surface battle	29
16	to reduce ability of enemy to plan & control units & tempo	30, 31, 32, 33
17	to destroy aircraft trying to penetrate airspace	34
18	to destroy enemy a/c trying to attack friendly forces	34
19	to avoid own detection	35, 36, 37
20	to minimize fuel usage	36, 37
21	to minimize uncertainty about environment	48, 49
22	to destroy the enemy's will to wage an effective air war	50

Goal Number	Goal	Included Goals
23	to neutralize enemy's will to wage an effective air war	50
24	to disrupt enemy's will to wage an effective air war	50
25	to negate surface based enemy air defenses	38, 39
26	to delay units not yet engaged by land forces	57
27	to disrupt units not yet engaged by land forces	57
28	to destroy units not yet engaged by land forces	57
29	to create opportunities for maneuver or advance of friendly forces	40, 41, 42
30	to divert combat and logistic assets to defend routes	43, 44, 45, 46
31	to delay buildup of combat strength	43, 44, 45, 46
32	to degrade efficiency with which assets can be used	43, 44, 45, 46
33	to deny enemy mobility	43, 44, 45, 46
34	to destroy threatening enemy aircraft	47
35	to navigate	90
36	to avoid threats	84, 85, 88, 90
37	to route plan	48
38	to negate enemy SAM air defense	51, 52, 53
39	to negate enemy AAA air defense	54, 55, 56
40	to protect the flank of friendly forces	57
41	to blunt enemy offensive maneuvers	57
42	to protect the rear of surface forces during retrograde maneuvers	57
43	to destroy enemy potential before it can effectively be used against friendly forces	58, 59, 60, 61
44	to disrupt enemy potential before it can effectively be used against friendly forces	62, 63, 64, 65
45	to divert enemy potential before it can effectively be used against friendly forces	66, 67, 68, 69
46	to delay enemy potential before it can effectively be used against friendly forces	70, 71, 72, 73
47	to intercept threatening enemy aircraft	75
48	to maintain currency of enemy's order of battle	74
49	to assess state of enemy readiness	74
50	to neutralize/destroy enemy aerospace forces	75
51	to neutralize SAM air defense	76
52	to degrade SAM air defense	76
53	to destroy SAM air defense	76
54	to neutralize AAA air defense	77
55	to degrade AAA air defense	77
56	to destroy AAA air defense	77
57	to target particular enemy equipment	75
58	to destroy enemy surface forces	80
59	to destroy enemy movement networks	80
60	to destroy enemy C3 networks	80
61	to destroy enemy combat supplies	80

Goal Number	Goal	Included Goals
62	to disrupt enemy surface forces	81
63	to disrupt enemy movement networks	81
64	to disrupt enemy C3 networks	81
65	to disrupt enemy combat supplies	81
66	to delay enemy surface forces	82
67	to delay enemy movement networks	82
68	to delay enemy C3 networks	82
69	to delay enemy combat supplies	82
70	to divert enemy surface forces	83
71	to divert enemy movement networks	83
72	to divert enemy C3 networks	83
73	to divert enemy combat supplies	83
74	to collect intelligence	75
75	to engage enemy targets	78, 79, 88
76	to physically attack SAM air defense	79
77	to electronically attack AAA air defense	79
78	to id all detected targets	84, 85, 86
79	to detect threats	90
81	to target a particular enemy surface force	87
81	to target a particular enemy movement network	87
82	to target a particular enemy C3 network	87
83	to target particular enemy combat supplies	87
84	to id enemy targets	89
85	to id neutral targets	89
86	to id friendly targets	89
87	to detect a enemy ground target	90
88	to track all detected targets	
89	to id targets	
90	to search for enemy targets	

List of References

List of References

[1] E. A. Hansen, "Cost-Effective Sensing During Plan Execution," *Proceedings of Twelfth National Conference on Artificial Intelligence*, vol. 2, July 31 - August 4 1994, pp. 1029-1035.

[2] V. J. Lumelsky, S. Mukhopadhyay, and K. Sun, "Dynamic Path Planning in Sensor-Based Terrain Acquisition." *IEEE Transactions on Robotics and Automation*, vol. 6, no. 4, pp. 462-472, August 1990.

[3] A. Stopp and T. Riethmüller, "Fast Reactive Path Planning By 2D And 3D Multi-Layer Spatial Grids For Mobile Robot Navigation," *Proceedings of the 1995 IEEE International Symposium on Intelligent Control*, Monterey, CA, August 27-29 1995, pp. 545-550.

[4] J. S. Zelek and M. D. Levine, "Teleo-Reactive Autonomous Mobile Navigation," *1995 Canadian Conference on Electrical and Computer Engineering*, Montreal, Quebec, Canada, September 5-8 1995, pp. 477-480.

[5] S. Wood, *Planning and Decision Making in Dynamic Domains,* New York, NY: Ellis Horwood, 1993.

[6] C-A. Brunet, R. Gonzalez-Rubio, and M. Tétreault, "A Multi-Agent Architecture for a Driver Model for Autonomous Road Vehicles," *1995 Canadian Conference on Electrical and Computer Engineering*, , Montreal, Quebec, Canada, September 5-8 1995, pp. 772-775.

[7] C-Y. Chong and M. Liggins II, "Fusion Technologies for Drug Interdiction," *Proceedings of the 1994 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI '94)*, Las Vegas, NV, October 2-5 1994, pp. 435-441.

[8] R. A. Adrian, "Sensor Management," *Proceedings of the 1993 AIAA/IEEE Digital Avionics System Conference*, Ft. Worth, TX, October 25-28 1993, pp. 32-37.

[9] S. Musick and R. Malhotra, "Chasing The Elusive Sensor Manager," *Proceedings of the IEEE 1994 National Aerospace and Electronics Conference, NAECON 1994*, vol. 1, Dayton, OH, May 23-27 1994, pp. 606-613.

[10] R. V. Denton, E. I. Alcaraz, J. Llinas, and K. J. Hintz, "Towards Modern Sensor Management Systems," in *Science of Command and Control: Part III Coping With Change*, Fairfax, VA: AFCEA International Press, 1994, pp. 118-134.

[11] E. Alcaraz, R. V. Denton, J. Llinas, and K. J. Hintz, *Formal Mathematical Methods for Sensor Management*, report prepared for the Wright Laboratory, Wright-Patterson AFB, OH, December 1992.

[12] M. L. Van Creveld, *Command in War*, Cambridge, MA: Harvard University Press, 1985.

[13] H. Van Trees, "Overview of Command and Control," *Principles of C3I Part I class notes*, August 1994.

[14] R. Malhotra, "Temporal Considerations in Sensor Management," *Proceedings of the IEEE 1995 National Aerospace and Electronics Conference, NAECON 1995*, vol. 1, Dayton, OH, May 22-26 1995, pp. 86-93.

[15] J. A. Simpson and E. S. C. Weiner (eds), *The Oxford Dictionary*, Oxford: Clarendon Press, 1989.

[16] J. M. Manyika and H. F. Durrant-Whyte. *Data Fusion and Sensor Management : A Decentralized Information-Theoretic Approach*, New York, NY: Ellis Horwood, 1994.

[17] R. Popoli, "The Sensor Management Imperative," in *Multitarget-Multisensor Tracking: Applications and Advances*, vol. II, Norwood, MA: Artech House Publishers, 1992.

[18] P. L. Rothman and S. G. Bier, "Evaluation of Sensor Management Systems," *Proceedings of the IEEE 1989 National Aerospace and Electronics Conference, NAECON 1989*, vol. 4, Dayton, OH, May 22-26 1989, pp. 1747-1752.

[19] Z. Zhang and K. J. Hintz, "OGUPSA Sensor Scheduling Architecture and Algorithm," *Signal Processing, Sensor Fusion, and Target Recognition V. Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 2755, Orlando, FL, April 8-10 1996, pp. 296-303.

[20] G. A. McIntyre and K. J. Hintz, "Sensor Measurement Scheduling: An Enhance Dynamic, Preemptive Algorithm," *Optical Engineering*, vol. 37, no. 2, pp. 517-523, February 1998.

[21] G. A. McIntyre and K. J. Hintz, "Sensor management simulation and comparative study," *Signal Processing, Sensor Fusion, and Target Recognition VI. Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 3068, Orlando, FL, April 20-25 1997.

[22] D. M. Buede and E. L. Waltz, "Issues in Sensor Management," *Proceedings. 5th IEEE International Symposium on Intelligent Control*, vol. 2, Philadelphia, PA, September 5-7 1990, pp. 839-842.

[23] B. McBryan, S. M. Meyer, and M. D. Knabach, "A Sensor Management Expert System for Multiple Sensor Integration (MSI)," *Proceedings of the IEEE 1990 National Aerospace and Electronics Conference, NAECON 1990*, vol. 3, Dayton, OH, May 21-25 1990, pp. 1107-1112.

[24] G. A. Kuperman, "Operator Interface For a Multi-sensor Target Acquisition System," *Proceedings of the IEEE 1994 National Aerospace and Electronics Conference, NAECON 1994*, vol. 2, Dayton, OH, May 23-27 1994, pp. 638-645.

[25] B. D. Leon and P. R. Heller, "An Expert System and Simulation Approach for Sensor Management and Control in Distributed Surveillance Network," *Applications of Artificial Intelligence V. Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 786, Orlando FL, May 18-20 1987, pp. 41-51.

[26] A. D. Pisano, and H. L. Jones, "An Expert Systems Approach to Adaptive Tactical Navigation," *The First Conference on Artificial Intelligence Applications,* Denver, CO, December 5-7 1984, pp. 460-464.

[27] R. Popoli and S. Blackman, "Expert System Allocation for the Electronically Scanned Antenna Radar," *Proceedings of the 1987 American Control Conference,* vol. 3, Minneapolis, MN, June 10-12 1987, pp. 1821-1826.

[28] B. Hersen, A. Lalo, D. Laneuville, and M. Mariton, "Image Based Target Tracking: Fusion Architecture and Sensor Management," *Signal Processing, Sensor Fusion, and Target Recognition II. Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 1955, Orlando, FL, April 12-14 1993, pp. 165-171.

[29] J. M. Nash, "Optimal Allocation of Tracking Resources," *Proceedings of the 1977 IEEE Conference on Decision and Control*, vol. 1, New Orleans, LA, December 7-9 1977, pp. 1177-1180.

[30] R. Fung, E. Horvitz, and P. Rothman, "Decision Theoretic Approaches to Sensor Management," DTIC # AB B172227, Wright-Patterson AFB, OH, Feb 1993.

[31] J. M. Manyika and H. Durrant-Whyte, "On Sensor Management in Decentralized Data Fusion," *Proceedings of the 31st Conference on Decision and Control*, vol. 4, Tucson, AZ, December 16-18 1992, pp. 3506-3507.

[32] A. Gaskell and P. Probert, "Sensor Models and a Framework for Sensor Management," *Sensor Fusion VI. Proceedings of the SPIE - The International Society for Optical Engineering,* vol. 2059, Boston, MA, September 7-8 1993, pp. 2-13.

[33] R. Washburn, A. Chao, D. Castanon, D. Bertsekas, R. Malhotra, "Stochastic Dynamic Programming for Far-Sighted Sensor Management," *1997 IRIS National Symposium on Sensor and Data Fusion*, 1997.

[34] J. M. Molina López, F. J. Jiménez Rodríguez, and J. R. Casar Corredera, "Fuzzy Reasoning For Multisensor Management," *1995 IEEE International Conference on Systems, Man, and Cybernetics*, vol. 2, Vancouver, British Columbia, Canada, October 22-25 1995, pp. 1398-1403.

[35] J. Zhongliang, X. Hong, and Z. Xueqin, "Information Fusion and Tracking of Maneuvering Targets with Artificial Neural Networks," *Proceedings of the 1994 IEEE International Conference on Neural Networks*, vol. 5, Orlando, FL, June 27 - July 2 1994, pp. 3403-3408.

[36] T. A. Brownell, "Neural Networks for Sensor Management and Diagnostics," *Proceedings of the IEEE 1994 National Aerospace and Electronics Conference, NAECON 1992*, vol. 3, Dayton, OH, May 18-22 1992, pp. 923-929.

[37] K. J. Hintz and E. S. McVey, "Multi-Process Constrained Estimation," *IEEE Transaction on Systems, Man, and Cybernetics*, vol. 21, no. 1, pp. 434-442, Jan/Feb 1991.

[38] C. E. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, vol. 27, no. 3, pp.379-423, July 1948.

[39] K. J. Hintz, "A Measure of the Information Gain Attributable to Cueing," *IEEE Transaction on Systems, Man, and Cybernetics*, vol. 21, no. 2, pp. 237-244, March/April 1991.

[40] G. A. McIntyre and K. J. Hintz, "An Information Theoretic Approach to Sensor Scheduling," *Signal Processing, Sensor Fusion, and Target Recognition V. Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 2755, Orlando, FL, April 8-10 1996, pp. 304-312.

[41] S. Kullback and R. A. Leibler, "On Information and Sufficiency," *The Annals of Mathematical Statistics*, vol. 22, pp. 79-86.

[42] W. Schmaedeke, "Information Based Sensor Management," *Signal Processing, Sensor Fusion, and Target Recognition II. Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 1955, Orlando, FL, April 12-14 1993, pp. 156-64.

[43] K. Kastella, "Event-Averaged Maximum Likelihood Estimation and Mean-Field Theory in Multitarget Tracking," *IEEE Transactions on Automatic Control,* vol. 40, no. 6, pp.1070-74, June 1995.

[44] K. Kastella, "A Maximum Likelihood Estimator for Report-To-Track Association," *Signal and Data Processing of Small Targets1993. Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 1954, Orlando, FL, April 12-14 1993, pp. 386-93.

[45] W. Schmaedeke and K. Kastella, "Event-averaged Maximum Likelihood Estimation and Information Based Sensor Management," *Signal Processing, Sensor Fusion, and Target Recognition III. Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 2232, Orlando, FL, April 4-6 1994, 91-96.

[46] K. Kastella, "Discrimination Gain to Optimize Detection and Classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, vol. 27, no. 1, pp. 112-116, January 1997. K. Kastella, "Discrimination Gain to Optimize Detection and Classification," *Signal and Data Processing, of Small Targets 1995: Proceedings of the SPIE -*

The International Society for Optical Engineering, vol. 2561, San Diego, CA, July 11-13 1995, pp. 66-70.

[47] K. Kastella, "Discrimination Gain for Sensor Management in Multitarget Detection and Tracking," *IEEE-SMC and IMACS Multiconference CESA '96*, vol. 1, Lille France, July 9-12 1996, pp. 167-172.

[48] K. Kastella and S. Musick, "The Search for Optimal Sensor Management," *Signal and Data Processing, of Small Targets 1996: Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 2759, Orlando, FL, April 9-11 1996, 318-29. K. Kastella and S. Musick, "Comparison of Sensor Management Strategies for Detection and Classification," *9th National Symposium on Sensor Fusion*, Monterey, CA, March 1996.

[49] E. Waltz and J. Llinas, *Multisensor Data Fusion*, Norwood, MA: Artech House, 1990.

[50] N. Upton and K Wallace, "Sensor Management Systems," Proceedings: Aerotech95 (IMechE / IEE / RAeS / SBAC conference), 1995.

[51] F. White, O. Kessler, and D. Noble, :Data Fusion Model," *Proceedings Sixth Joint Service Data Fusion Symposium*, Laurel MD, June 14-18 1993, pp. 49-76.

[52] M. E. Liggins II and A. Bramson, "Off-Board Augmented Fusion for Improved Target Detection and Track," *Proceedings of the 27th Asilomar Conference on Signals, Systems, and Computers*, vol. 1, Pacific Grove, CA, November 1-3 1993, pp. 295-299.

[53] J. Marsh, S. Ruberg, C. Boettcher, E. Trujillo, S. Oxford, "Object-oriented Management of Real-time Data in Integrated Avionics Architectures," *Proceedings of the IEEE 1993 National Aerospace and Electronics Conference, NAECON 1993*, vol. 1., Dayton, OH, May 24-28 1993, pp. 606-613.

[54] A. V. Forman, Jr., D. B. Brown, J. H. Hughen, R. R. Pressley, A. R. Sanders, and D. J. Sullivan, "Multisensor Target Recognition System (MUSTRS)," *Proceedings of the 27th Asilomar Conference on Signals, Systems, and Computers*, vol. 1, Pacific Grove, CA, November 1-3 1993, pp. 263-267.

[55] P. M. Lynch and J. De Paso, 'An Object Oriented Intelligent Control Architecture," *Proceedings of the 1992 American Control Conference*, vol. 3, Chicago, IL, June 24-26 1992, pp. 2295-2301.

[56] S. G. Bier, P. L. Rothman, and R. A. Manske, "Intelligent Sensor Management For Beyond Visual Range Air-to-Air Combat," *Proceedings of the IEEE 1988 National Aerospace and Electronics Conference, NAECON 1988*, vol. 1, Dayton OH, May 23-27 1988, pp. 264-269.

[57[P. Rothman and S. Bier, "Intelligent Sensor Management Systems for Tactical Aircraft," *Second National Symposium on Sensors and Sensor Fusion*, Chicago IL, March 1989.

[58] D. Buede, J. Martin, and J. Sipos, "AIFSARA: A Test Bed for Fusion, Situation Assessment and Response," *Proceedings of the 4th AAAIC 1988 Conference*, Dayton, OH, October 1988, pp. 335-344.

[59] P. Rohde and L. Jamerson, "Integration of Multisensor Fusion and Sensor Management," *Data Fusion Symposium*, 1990.

[60] R. Popoli, S. Blackman, and T. Broida , "A Utility Theory Method for Allocation of the Radar Electronically Scanned Antenna," Hughes Technical Report, 1989.

[61] J. Llinas, K. Hintz, and B. Beale, "Applications of Automatic Control Theory to Sensor Scheduling," report prepared for Naval Air Warfare Center, Warminster, PA, 1992.

[62] K. A. Dunkelberger, "Hypothesis-driven Distributed Sensor Management," *Signal Processing, Sensor Fusion, and Target Recognition III: Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 2232, Orlando, FL, April 4-6 1994, pp. 79-90.

[63] G. Hager and M. Mintz, "Estimation Procedures for Robust Sensor Control, *Uncertainty in Artificial Intelligence*, vol. 3, North-Holland: Elsevier Science Publishers, 1989, pp. 285-300.

[64] S. A. Hutchinson and A. C. Kak, "Planning Sensing Strategies in a Robot Work Cell with Multi-sensor Capabilities," *IEEE Transaction on Robotics and Automation*, vol. 5, no. 6, pp. 765-783, December 1991.

[65] B D'Ambrosio and R. Fung, "Far-Sighted Decision Theoretic Sensor Management," report prepared for Wright Laboratory, Wright-Patterson AFB, OH, 1996.

[66] M. Cone and S. Musick, "Scheduling for Sensor Management Using Genetic Search," *Proceedings of the IEEE 1996 National Aerospace and Electronics Conference, NAECON 1996*, vol. 1, Dayton OH, May 20-23 1996, pp. 150-156.

[67] K. J. Hintz and Z. Zhang, "Dynamic Resource Allocation and Sensor Management for OBATS," report prepared for Grumman Aerospace Corporation, 1993.

[68] J. M. Manyika and H. F. Durrant-Whyte, "An Information-Theoretic Approach to Management in Decentralized Data Fusion." *Sensor Fusion V. Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 1828, Boston, MA, November 15-17 1992, pp. 202-213.

[69] P. Greenway and R. Deaves, "Sensor Management Using the Decentralized Kalman Filter," *Sensor Fusion VII. Proceedings of the SPIE - The International Society for Optical Engineering,* vol. 2355, Boston, MA, October 31 - November 1 1994, pp. 216-225.

[70] P. Greenway and R. Deaves, "An Information Filter For Decentralized Sensor Management," *Signal Processing, Sensor Fusion, and Target Recognition III. Proceedings of the SPIE - The International Society for Optical Engineering,* vol. 2232, Orlando, FL, April 4-6 1994, pp. 70-78.

[71] T. Jenison, "Improved Data Structures, Multiple Targets/Target Types and Multiple Sensors in Discrimination Based Sensor Management," Master's Thesis, University of Minnesota, July 1996.

[72] R. C. James and G. James, *Mathematics Dictionary (5th ed)*, New York: Chapman & Hall, 1992.

[73] J. R. Cloutier, C-F Lin, and C. Yang, "Enhance Variable Dimension Filter for Maneuvering Target Tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 3, pp. 786-796, July 1993.

[74] P. R. Halmos, *Naïve Set Theory*, Princeton NJ: D. Van Nostrand Company, Inc., 1960.

[75] M. W. Shields, *An Introduction to Automata Theory*, Oxford, England: Blackwell Scientific Publications, 1987.

[76] T. Donnellan, *Lattice Theory*, Oxford, England: Peragamon Press, 1968

[77] "NASA Strategic Plan, Appendix 2: Summary and Relationship of Mission, Long-Term," Oct 30, 1997, <http://www.hq.nasa.gov/office/nsp/appendix.htm> (Dec 19, 1997).

[78] "Seminar/Correspondence Lesson Book," Volume 4 (AY94), Air Command and Staff College, Maxwell AFB, AL, 1994.

[79] R.G. Almond, *Graphical Belief Modeling*, London: Chapman and Hall, 1995.

[80] K. Hoffman and R. Kunze, *Linear Algebra,* Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1971.

[81] J. Gilbert and L. Gilbert, *Linear Algebra and Matrix Theory,* San Diego, CA: Academic Press, 1995.

[82] B-S. K. Skagerstam, "On The Notions of Entropy and Information," *Journal of Statistical Physics*, vol. 12, no. 6, pp. 449-462, June 1975.

[83] P. Kalata and F. Priemer, "Linear Prediction, Filtering, and Smoothing: An Information-Theoretic Approach," *Information Sciences*, vol. 17, no. 1, pp. 1-14, February 1979.

[84] A. Gelb (editor), *Applied Optimal Estimation*, Cambridge, Ma: M.I.T. Press, 1974

[85] Y. Tomita, S. Ohmatsu, and T. Soeda, "An Application of Information Theory to Estimation Problems,: *Information and Control*, vol. 32, no. 2, pp. 101-111, October 1976.

[86] Y Tomita, S. Omatu, and T. Soeda, "An Application of Information Theory to Filtering Problems,: *Information Sciences*, vol. 11, no. 1, 13-27, 1976.

[87] I. Y. Hoballah and R. K. Varshney, "An Information Theoretic Approach to the Distributed Detection Problem," *IEEE Transactions on Information Theory*, vol. 35, no. 5, pp. 988-994, September 1989.

[88] L. G. Clark, L. I. Perlovsky, W. H. Schoendorf, C. P. Plum, T. L. Keller, "Evaluation of FLIR Sensors For Automatic Target Recognition Using An Information-Theoretic Approach," *Automatic Object Recognition II, Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 1700, Orlando, FL, April 22-24 1992, pp. 423-434.

[89] L. G. Clark, L. I. Perlovsky, W. H. Schoendorf, C. P. Plum, T. L. Keller, "Evaluation of Forward-Looking Infrared Sensors For Automatic Target Recognition Using An Information-Theoretic Approach," *Optical Engineering*, vol. 31, no. 12, pp. 2618-2627, December 1992.

[90] L. G. Clark, L. I. Perlovsky, W. H. Schoendorf, C. P. Plum, T. L. Keller, "Evaluation of FLIR Sensors For Automatic Target Recognition Using An Information-Theoretic Approach," *IEEE 1993 National Aerospace and Electronics Conference NAECON 1993*, vol. 2, Dayton, OH, May 24-28 1993, pp. 1048-1055.

[91] E. Soofi, "Capturing the Intangible Concept of Information," *Journal of the American Statistical Association*, vol. 89, no. 428, pp. 1243-1254, December 1994.

[92] R. M. Turner and A. W. Bridgewater, "An Information-Theoretic Approach To Energy Management For Surveillance By Electronically Agile Radars, " *Sixteenth Asilomar Conference on Circuits, Systems, and Computers*, Pacific Grove, CA, November 8-10 1982, pp. 230-234.

[93] P. Greenway, R. H. Deaves, and D. R. Bull, "Communications Management in Decentralised Data Fusion Systems,: *1996 IEEE/SICE RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Washington, D.C., December 8-11 1996, pp. 796-805.

[94] M. G. Oxenham, D. J. Kewley, M. J. Nelson, "Measure of Information For Mulit-level Data Fusion," *Signal Processing, Sensor Fusion, and Target Recognition V. Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 2755, Orlando, FL, April 8-10 1996, pp. 271-82.

[95] W. H. Barker II, "Information Theory and the Optimal Detection Search," *Operations Research*, vol. 25, no. 2, pp. 304-314, March-April 1977.

[96] E. S. Soofi, "Information Theoretic Regression Methods," in *Advances in Econometrics: Applying Maximum Entropy to Econometric Problems*, vol. 12, pp. 25-83, T. B. Fomby and R. C. Hill (eds.), Greenwich, Ct: JAI Press, 1997.

[97] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association,* Orlando FL: Academic Press, Inc., 1988.

[98] K. Spingarn and H. L. Weidemann, "Linear Regression Filtering and Prediction for Tracking Maneuvering Aircraft Targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-8, no. 6, pp. 800-810, November 1972.

[99] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, Englewood Cliffs, NJ: Prentice-Hall Inc., 1979.

[100] F. R. Castella and F. G. Dunnebacke, "Analytical Results for the *x, y* Kalman Tracking Filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-10, no. 6, pp. 891-895, November 1974.

[101] S. N. Balakrishnan and J. L. Speyer, "Coordinate-Transformation-Based Filter for Improved Target Tracking," *Journal of Guidance, Control, and Dynamics,* vol. 9, no. 6, pp. 704-709, November/December 1986.

[102] Singer, R. A., "Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-6, no. 4, .pp. 473-483, July 1970.

[103] Singer, R. A. and K. W. Behnke, "Real-Time Tracking Filter Evaluation and Selection for Tactical Applications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-7, no. 1, pp. 100-110, January 1971.

[104] Singer, R. A., R. G. Sea, and K. B. Housewright, "Derivation and Evaluation of Improved Tracking Filters for Use in Dense Multitarget Environments", *IEEE Transactions on Information Theory*, vol. IT-20, no. 4, pp. 423-432, July 1974.

[105] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice*, Englewood Cliffs, NJ: Prentice-Hall, Inc., 1993.

[106] J. Choi, "Comparison of Sklansky and Singer Tracking Models Via Kalman Filtering," *IEEE Southeastcon '83*, Orlando, FL, April 11-14 1983, pp. 335-338.

[107] J. P. Helferty, "Improved Tracking of Maneuvering Targets: The Use of Turn-Rate Distributions for Acceleration Modeling,' *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 4, pp.1355-1361, October 1996.

[108] D. Lerro and Y. Bar-Shalom, "Tracking with Debiased Consistent Converted Measurements Versus EKF," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, n. 3, pp. 1015-1022, July 1993.

[109] M. S. Woolfson, "An Evaluation of Manoeuver Detector Algorithms," *The GEC Journal of Research Incorporating Marconi Review*, vol. 3, no. 3, pp. 181-190, 1985.

[110] Y. Bar-Shalom and K. Birmiwal, "Variable Dimension Filter for Maneuvering Target Tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-18, no. 5, pp. 621-628 September 1982.

[111] M. Efe and D. P. Atherton, "Maneuvering Target Tracking Using Adaptive Turn Rate Models in The Interacting Multiple Model Algorithm," *Proceedings of the 35th Conference on Decision and Control*, vol. 2, Kobe, Japan, December 11-13 1996, pp. 3151-3156.

[112] W. D. Blair, G. A. Watson, and T. R. Rice, "Tracking Maneuvering Targets With An Interacting Multiple Model Filter Containing Exponentially-Correlated Acceleration Models," *Proceedings of the Twenty-third Southeastern Symposium on System Theory*, Columbia, SC, March 10-12 1991, pp. 224-228.

[113] H. L. Van Trees, *Detection, Estimation, and Modulation Theory*, New York, NY: John Wiley and Sons, Inc., 1968.

Curriculum Vita

Gregory A. McIntyre was born on September 20, 1959, in Topeka, Kansas, and is an American citizen. He graduated from Washburn Rural High School, Topeka, Kansas, in 1979. He received a Bachelor of Arts in Math and Computer Science from Washburn University of Topeka and was commission as a 2nd Lieutenant in the Air Force through ROTC in 1981. He also holds a Master of Science in Operations Research from the Air Force Institute of Technology (AFIT) where he was a Distinguished Graduate. Greg is a Lieutenant Colonel in the United States Air Force and was on an AFIT doctoral fellowship while attending George Mason University. Upon completion of his fellowship, he was be assigned to AFIT as an Assistant Professor of Operations Research.